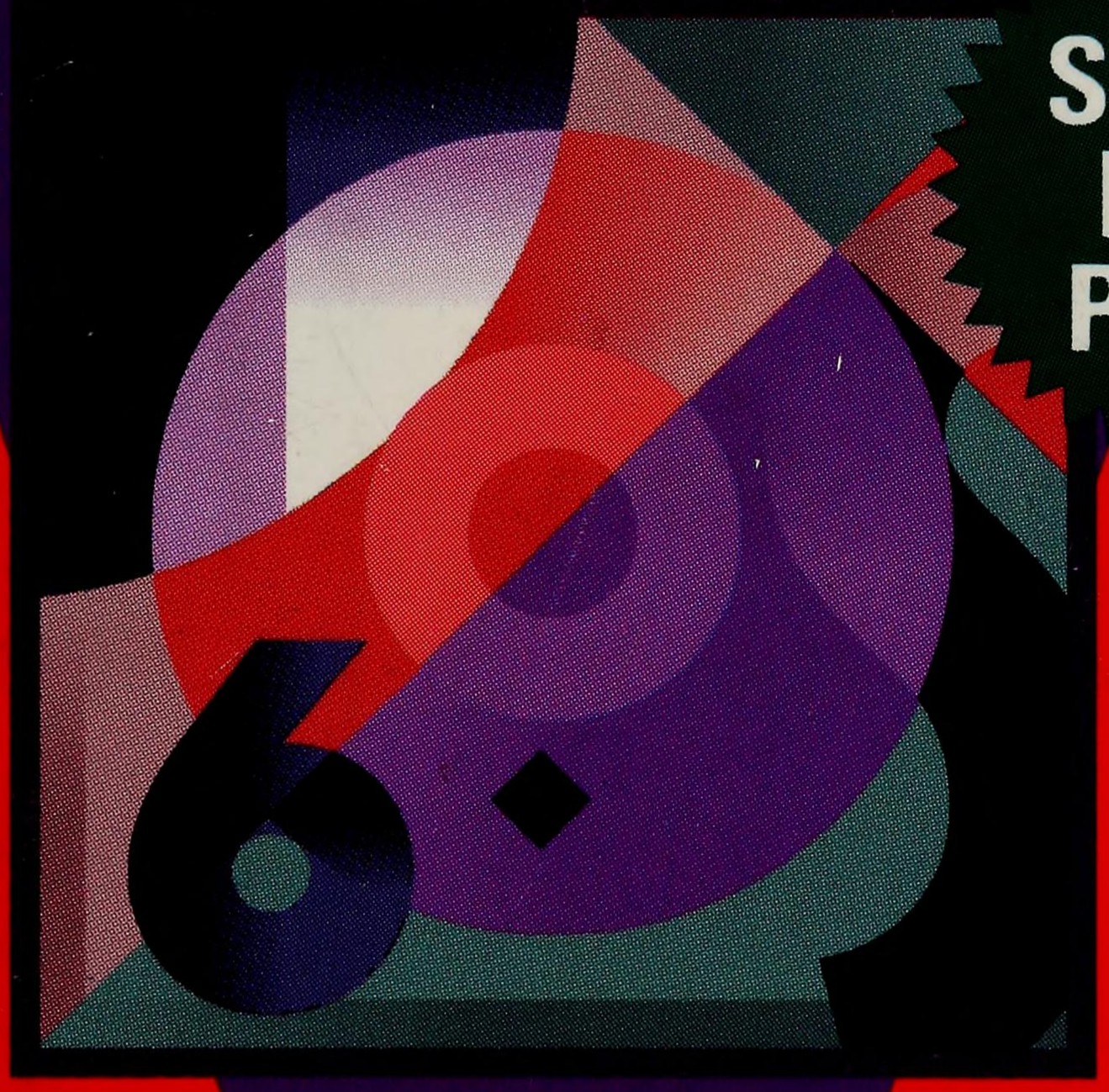
RANDOM HOUSE
ELECTRONIC PUBLISHING

FRIENDLY

DOS 6

2ND EDITION



SUPER
LOW
PRICE

THE LEBLOND GROUP

BASIC INFORMATION FOR THE NEW USER

DO NOT REMOVE
CARDS FROM POCKET

ALLEN COUNTY PUBLIC LIBRARY
FORT WAYNE, INDIANA 46802

You may return this book to any agency, branch,
or bookmobile of the Allen County Public Library.

DEMCO

WAY

ALLEN COUNTY PUBLIC LIBRARY
3 1833 02457 1736

WITHDRAWN

005.446 M879fa

JUN 2 7 1994

Friendly DOS 6

Second Edition

Friendly Computer Books are available for the following major software programs:

Friendly DOS 6
by The LeBlond Group

Friendly Access
by Douglas Hergert

Friendly Macintosh
by Kay Yarborough Nelson

Friendly PCs
by Mary Campbell

Friendly Excel 4.0 for Windows
by Jack Nimersheim

Friendly Quicken for Windows
by The LeBlond Group

Friendly Windows 3.1
by Kay Yarborough Nelson

Friendly Word for Windows
by Jack Nimersheim

Friendly WordPerfect
by Kay Yarborough Nelson

Friendly DOS 6

Second Edition

The LeBlond Group

RANDOM HOUSE
ELECTRONIC PUBLISHING

Friendly DOS 6, Second Edition

Copyright © 1993 by The LeBlond Group

Produced by MicroText Productions
Composed by Context Publishing Services

All rights reserved. No part of the contents of this book may be reproduced in any form or by any means without the written permission of the publisher.

Published in the United States by Random House, Inc., New York, and simultaneously in Canada by Random House of Canada, Limited.

Manufactured in Canada.

0 9 8 7 6 5 4 3 2 1

First edition

ISBN 0-679-79190-6

Allen County Public Library
900 Webster Street
PO Box 2270
Fort Wayne, IN 46801-2270

The author(s) and publisher have used their best efforts in preparing this book, and the programs contained herein. However, the author(s) and publisher make no warranties of any kind, express or implied, with regard to the documentation or programs contained in this book, and specifically disclaim without limitation, any implied warranties of merchantability and fitness for a particular purpose with respect to program listings in the book and/or the techniques described in the book. In no event shall the author or publisher be responsible or liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential or any other damages in connection with or arising out of furnishing, performance, or use of this book or the programs.

Trademarks

A number of entered words in which we have reason to believe trademark, service mark, or other proprietary rights may exist have been designated as such by use of initial capitalization. However, no attempt has been made to designate as trademarks or service marks all personal computer words or terms in which proprietary rights might exist. The inclusion, exclusion or definition of a word or term is not intended to affect, or express any judgment on, the validity or legal status of any proprietary right which may be claimed in that word or term.

New York Toronto London Sydney Auckland

Contents

Preface xiii

1 Introducing DOS 1

Understanding DOS 1
Working with the DOS Shell 3

2 Working with Directories 11

The DOS Hierarchical Structure 11
Changing Drives 12
Changing Directories 13
Revealing and Hiding Subdirectories 16
Paths 17
Creating a Directory 18
Removing a Directory 18

3 Working with Files 21

Naming Files 21
Copying Files 22

4 More File Operations 29

Moving Files 29
Renaming Files 32
Renaming a File in Its Current Location 33
Deleting Files 35

5 Running Programs 37

Files that can be Run from DOS 37
Running a Program from the File List 39
Running a Program with the
File Run Command 40
Running a Program Included in a Group 42

6 Selecting Multiple Files for a Command 47

Methods of Selection 47

7 Different Ways of Viewing Files and Directories 53

Changing the DOS Shell Screen Arrangement 53

8 Controlling the File List 59

Changing File Display Options 59
Rereading All Files and Directories 64

9 Searching for Files 65

Searching for a Single File 65
Searching for a Group of Files 66

10 Building Program Groups 69
Understanding Program Groups 69
Working in the Program List Area 71
Creating a New Program Group 72
Adding Program Items to a Group 72
Deleting Program Groups and Items 76
Rearranging Program Groups and Items 76
Changing the Properties of a Program Item 77

11 Customizing the DOS Shell 79
Customizable Features 79
Changing Colors 79
Changing the Display Mode 80
Creating File Associations 83

12 Special File Operations 87
Changing File Attributes 87
Viewing a File's Contents 89

13 Using the MS-DOS Editor 93
Editing ASCII Files 93
Creating a Memo 94
Exiting the MS-DOS Editor 97

14 Formatting a Floppy Disk 99
Why Format? 99
Testing for Formatting 99
Formatting a New Disk 100
Creating a System Disk 104

15 Copying Floppy Disks 107
Understanding Disk Copy 107

16 Task Switching 111
About the Task Swapper 111
Starting More Than One Program 112
Switching Between Programs 114
Quitting Programs 115

17 Working at the DOS Command Line 119
The Purpose of the DOS Shell 119
Understanding COMMAND.COM 120
Using Commands 120
Using Doskey 124
Getting Help at the Command Line 125

18 Working with Drives and Directories 127
Changing to a Different Drive 127
Changing to a Directory 128
Viewing the Contents of a Directory 129
Creating Directories and Subdirectories 129
Viewing a Directory Tree 130
Removing Directories 131

19 Getting Information about Files 135
Getting Information about a Specific File 135
Getting Information about a Group of Files 136

Controlling the Files List 137
Viewing and Changing File Attributes 138

20 Working with Files from the Command Line 141

Renaming a File 141
Copying Files from the Command Line 142
Copying a Single File 143
Changing the Name of a File While Copying 143
Deleting Files 145
Moving Files 146

21 Running Programs from the Command Line 151

About Executable Files 151
Passing Parameters to Programs 152
Making the Leap from the DOS Shell 152

22 Command Line Shortcuts 155

Cutting Down on Typing 155
Current Location Shortcuts 155
Previous Location Shortcuts 157

23 Undeleting Files 159

Understanding Undelete 159
Using Undelete in the Current Directory 160
Using Undelete Outside the Current Directory 162
Selective Use of Undelete 162

24 Using Higher Levels of Delete Protection 163

About Delete Tracker and Delete Sentry 163
Enabling Delete Sentry Protection 165
Enabling Delete Tracker Protection 166
Setting Parameters for Undelete 166

25 Backing Up Files 169

Why Back Up? 169
Using Backup 170
Configuring Backup 170

26 Restoring Files 177

Understanding Restore 177
Restoring Files to Their Original Locations 177

27 Copying Entire Directories 181

The Xcopy Command 181
Copying a Single Directory 181
Copying Subdirectories 183

28 Working with ASCII Files 185

What is an ASCII File? 185
Viewing ASCII Files 185
Printing an ASCII File 186

29 Creating Batch Files 189

Understanding Batch Files 189
Creating a Simple Batch File 189
Running the Batch File 190

30 Modifying AUTOEXEC.BAT 193

The PC's Boot Sequence 193
Before You Proceed 194
Loading AUTOEXEC.BAT into MS-DOS Editor 196
AUTOEXEC.BAT Commands 197
Saving AUTOEXEC.BAT in the MS-DOS Editor 200

31 Increasing Disk Space Using DoubleSpace 201

Compressing Your Hard Disk 201
Setting Up DoubleSpace 202
Seeing the Effects of DoubleSpace 203

32 Defragmenting Your Hard Disk 205

The Problem with Fragmentation 205
Combatting Fragmentation with Defragmenter 205

33 Virus Detection 209

Protecting Your System from Viruses 209
What are Viruses? 209
Using Anti-Virus to Scan for Viruses 210

Epilogue: Where To Go from Here 213

Index 217

Preface

Most of us become computer users because we have to, because knowledge of a particular software package is needed for a job, or because computer-assisted productivity is essential to success in business. There are hundreds of reasons. Computers and software are only the means to an end. They have become a necessity of life, and this requirement shapes the way we go about learning how to use software.

Not everyone is interested in every detail of a particular program. Here is a quick, no-nonsense introduction that teaches the basic skills needed to use the software.

In approximately 216 pages, each Friendly Computer Book covers the basic features of a specific popular software in a way that will get new users up and running quickly. The result is a series of computer books that has these unifying characteristics:

- **Topic-oriented organization.** Short, self-contained lessons focus on a particular topic or area that is important in learning to use the software. When

you finish the lesson, you'll have mastered an aspect of the software.

- **Spacious layout.** Large type and a spacious layout make the books easy on the eyes and easy to use.
- **Step-by-step approach.** Numbered lists help you to concentrate on the practical steps needed to get your work done.
- **Numerous screen shots.** Each lesson contains at least two screen shots that show you exactly how your screen should look.
- **Frequent use of icons.** Many eye-catching icons—drawing attention to important aspects of the text and software—are placed throughout the book.
- **Lay-flat binding.** Friendly Computer Books stay open as you work.
- **And finally, a low, low price.**

For many users Friendly Computer Books are all they'll need. For others who want to learn more about the software, we've suggested further readings.

Enjoy the friendly approach of Friendly Computer Books!

◆ *Lesson* ◆

1

Introducing DOS

Understanding DOS

DOS is an acronym for "disk operating system," and it is an integrated collection of programs that acts as a link between you and the PC. As soon as you turn on your PC, DOS starts running and continues to run as long as the power remains on.

This book assumes you are using DOS 6. If you're running a version earlier than DOS 5, you won't be able to complete many of the lessons presented here. (If you're not sure which version of DOS you have, use the Ver command; see the example on the following page.)

DOS has three primary functions. First, it manages files and directories. A *file* is a collection of information stored in a PC. Files can contain programs, letters, lists, budgets, or anything else that you want to collect and store as a group. DOS has built-in tools for creating new files and adding to existing ones. These tools also let you copy,

move, erase, and rename files. Every file is part of some *directory*, which keeps files organized. If every file were located in the same place, it would take DOS a long time to search through all the files and find a particular one. Directories provide a way of addressing files so that DOS (and you) can access them quickly and easily.

Second, DOS runs programs. The "brains" of a PC consist of a microprocessor and RAM (random access memory—temporary memory that loses its contents when you turn off the PC's power). A program consists of a sequence of instructions that are placed, for the most part, one after the other in RAM. The microprocessor reads and executes each instruction in sequence. DOS controls the process that loads programs from disk into RAM and directs the microprocessor to execute them. You may have DOS run one of its many utility programs or a program supplied by a software developer, like Lotus 1-2-3.

Finally, DOS interacts with devices that are attached to your computer. DOS gives the PC access to devices such as the screen, keyboard, printers, disk drives, and CD-ROMs.

DOS isn't very flashy. When your PC first comes up, the only evidence that DOS is working is a command line that looks like this:

```
C:\>
```

Following the C:> prompt, you can enter commands that you want DOS to execute. Try this example:

1. Type **ver**.
2. Press **Enter**.

DOS displays a message similar to the one below showing its version number:

```
MS-DOS Version 6.00
```

Note: If someone else set up your PC for you, you may not see the DOS prompt when you first boot (start) your PC. Instead, you may see a menu system that lets you select various programs. Typically, such a menu system will offer an option for directly accessing DOS; Select that option.

Working with the DOS Shell

One of DOS's most useful utility programs is the DOS Shell. This program hides much of the complexity of working directly at the DOS command line and is a good place to start when you're first learning DOS.

To start the DOS Shell, type **dosshell** and press **Enter**. After a brief pause, your screen should look something like Figure 1.1.

If the DOS Shell doesn't load, try navigating to the DOS directory. Type cd \dos and press Enter. Then attempt the dosshell command again. If the DOS Shell still doesn't load, try another drive; for example, if the DOS prompt appears as D:>, type c: and press Enter.

Moving between Areas

When you first start the DOS Shell, your screen is divided into four main areas:

Drive icons area This area shows in icon form the disk drives that are available on your system. For example, Figure 1.1 shows a PC with two floppy

Drive icons area *(continued)*

disk drives (A and B) and two hard disk drives (C and H). By selecting a drive icon—for example, [A:]—you can make that drive current (see Lesson 2).

Directory tree area

This area shows the structure of the current drive in tree form. When you choose a new drive icon, the tree that you see here changes to match the structure of the drive whose icon you've chosen.

File list area

This area displays the files that reside in the directory currently chosen in the directory tree area.

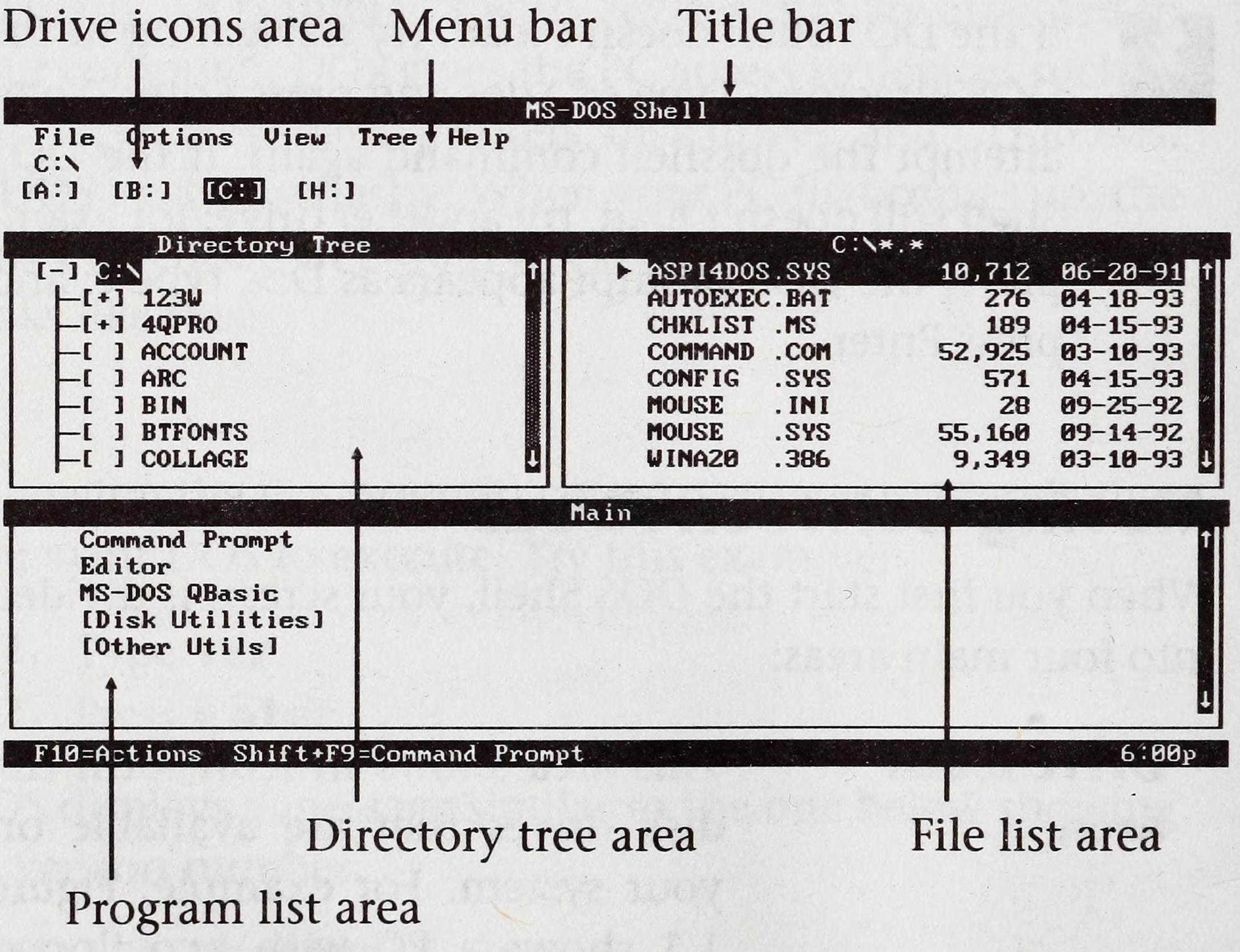


Figure 1.1 The DOS Shell

Program list area — This area shows the programs that are accessible directly from the DOS Shell.

Initially, the drive icons area is the active area, meaning that a drive icon (usually [C:]) is highlighted and any keys you press are directed toward that area of the screen. To move to another area using the keyboard, use Tab or Shift+Tab, as in this example:

1. To move from the drive icons area to the directory tree area, press **Tab**.
2. To move to the file list area, press **Tab** again.
3. To move back to the directory tree area, press **Shift+Tab**.

⇨ ***Note:*** If you have a mouse installed on your PC, you can use it to move around in the DOS Shell. To move to the program list area, for example, click on that area's title bar, the one that reads "Main."

Selecting Menu Commands

The menu bar near the top of the DOS Shell screen gives you access to many powerful commands. To activate the menu bar using the keyboard, press either **Alt** or **F10**.

Suppose you have a color screen and want to change the DOS Shell color scheme to something more lively, like Hot Pink. To do so, use the Options Colors command, as follows:

1. Press **Alt**. The **File** command appears highlighted in the menu bar. In addition, the first letter of each menu command is highlighted, meaning that you can select the command by typing that letter.

2. Press **o** to select the **Options** command, or use → to move to the Options command and press **Enter**. Figure 1.2 shows the pull-down menu that appears.
3. Select the **Colors** option by typing its highlighted letter, in this case, **o**; or use ↓ to move to the Colors option and press **Enter**. The DOS Shell presents a list of colors from which to choose.
4. Press ↓ repeatedly until **Hot Pink** is highlighted in the list. (If you go too far, press ↑ to move back up.)
5. Press **Enter** to confirm your choice. The colors of the DOS Shell title bar and highlights change to shades of pink.

To restore the DOS Shell to the default color scheme, repeat the procedure, this time choosing **Ocean** from the Colors list. To cancel a menu command at any time, press **Esc**.

Tip

The quickest way to select an option from the DOS Shell main menu is to press Alt+*Letter*, where *Letter* is the option's first letter—for example, press Alt+F for the File command.

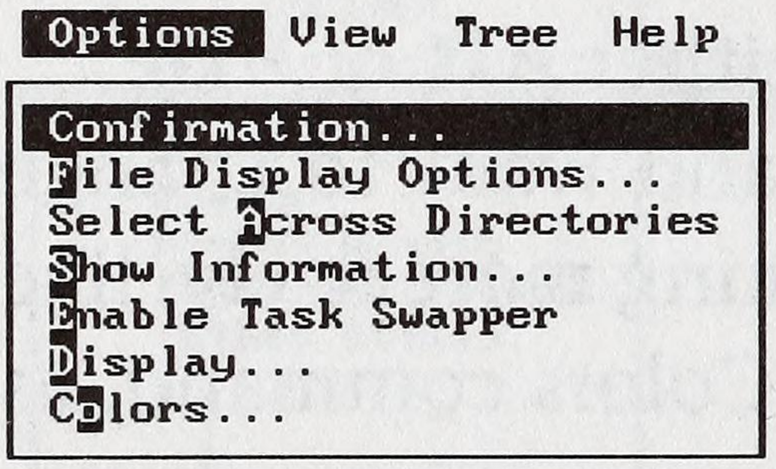


Figure 1.2 The Options pull-down menu

Note: To select a menu option using the mouse, simply click on it. To cancel a menu command, click on any area outside the menu.

As you explore the menu system, you'll discover that certain commands are grayed. This simply means that these commands are not available in the context you're currently in. For example, the Move option in the File menu appears grayed until you highlight a file in the file list area.

Using Dialog Boxes

A number of DOS Shell commands present a dialog box requesting information from you. Figure 1.3 shows an example. Typical elements contained in a dialog box are:

Text box Accepts text input. Table 1.1 shows the keys that you can use to edit the contents of a text box. If you have a mouse, click anywhere within the text box to begin editing.

Radio buttons These are grouped together to show you a fixed set of choices that are mutually exclusive. To select a radio button with the keyboard, use the ↑ or ↓ keys to move to it. If you have a mouse, click on a radio button to turn it on or off.

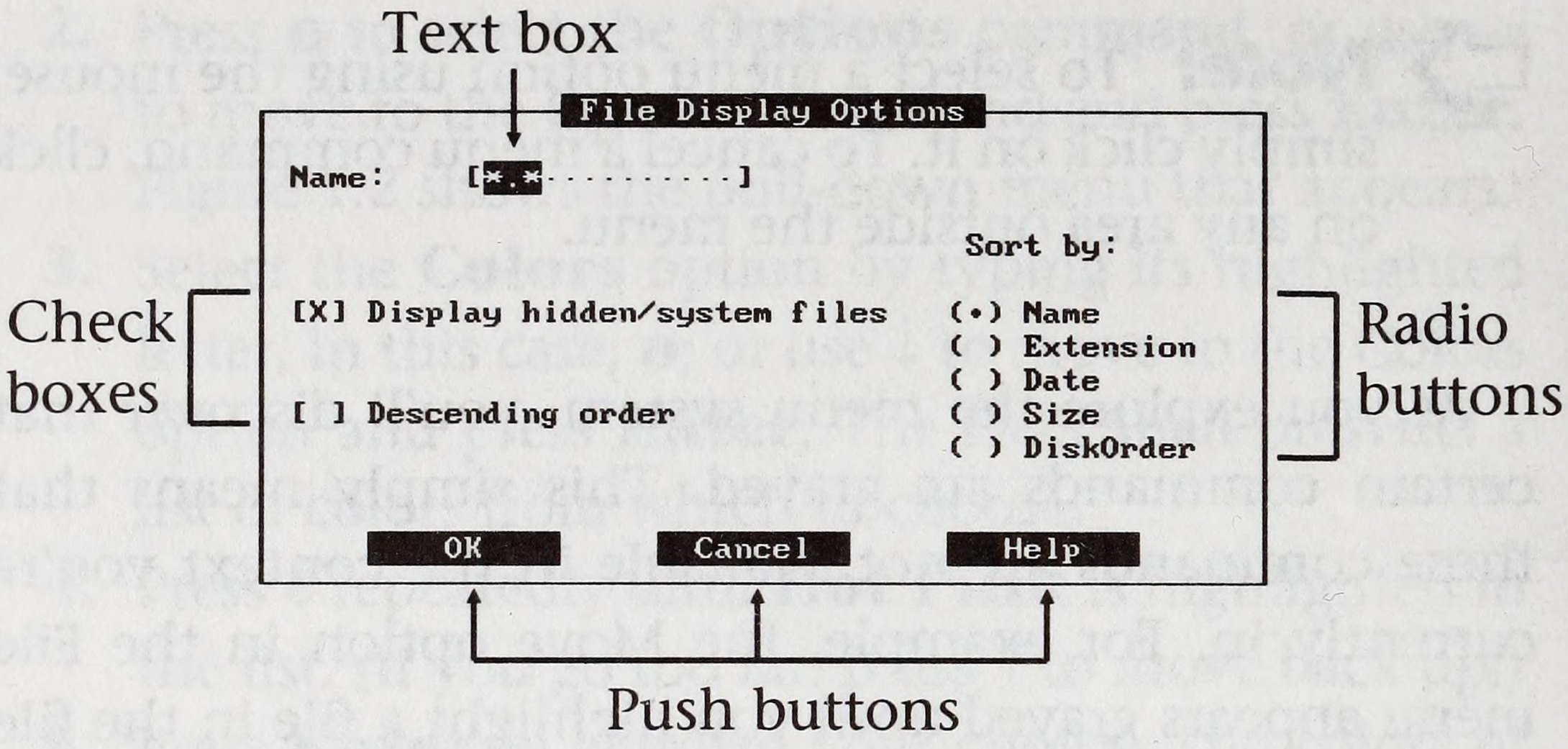


Figure 1.3 A sample dialog box

Check box Acts like a switch. An X appears in the check box when it is turned on. Unlike radio buttons, check boxes are not mutually exclusive. Press the **Spacebar** to turn the check box on and off; if you have a mouse, click on it to turn it on or off.

Push button Initiates an action. For example, the OK push button accepts the current settings and closes the dialog box (the same as pressing Enter), the Cancel push button closes the dialog box without changing settings (the same as pressing Esc), and the Help push button displays help about the dialog box.

To move from one item to the next in a dialog box, press the **Tab** key. When items are aligned vertically—for example, the two check boxes in Figure 1.3—you can also

Table 1.1 Keys for Editing

Key	Function
→	Moves right one character.
←	Moves left one character.
Home	Moves to the start of the text entry.
End	Moves to the end of the text entry.
Del	Deletes the character at the cursor (or the highlighted characters).
Shift	When used with the → and ← keys, lets you highlight a group of characters.

use ↑ and ↓ to move between them. When items are aligned horizontally—for example, the OK, Cancel, and Help buttons in Figure 1.3—you can use → and ← to move among them.

Exiting the DOS Shell

To leave the DOS Shell and return to the DOS command line, use the **File Exit** command, or its shortcut, **Alt+F4**. If you've customized the DOS Shell's configuration in any way (see Lesson 11), it saves the configuration settings to disk so that it can use them again the next time you start the program.

3 1833 02457 1736

◆ *Lesson* ◆

2

Working with Directories

The DOS Hierarchical Structure

A disk drive's contents consist of a hierarchical structure of files and directories. You can think of this structure as an inverted tree like the one in Figure 2.1. At the top is a single directory, called the *root directory* or \ *(backslash) directory*. This is analogous to the trunk of the tree. All other directories branch out from the root, just as the limbs of a tree branch out from its trunk. At the bottom of the structure are files, which are like the leaves of a tree. In between the root directory and the files are other directories. These can be thought of as the branches of a tree: They lead from the root directory to other directories and files.

Figure 2.1 shows the structure of a typical hard drive. The ellipses (...) under the names of directories indicate that these directories may branch out into other subdirectories, which for reasons of space are not shown. Any directory that is inside another is called a *subdirectory*. In Figure 2.1, for example, BUSINESS and PERSONAL are subdirectories of LETTERS. Any directory that contains a

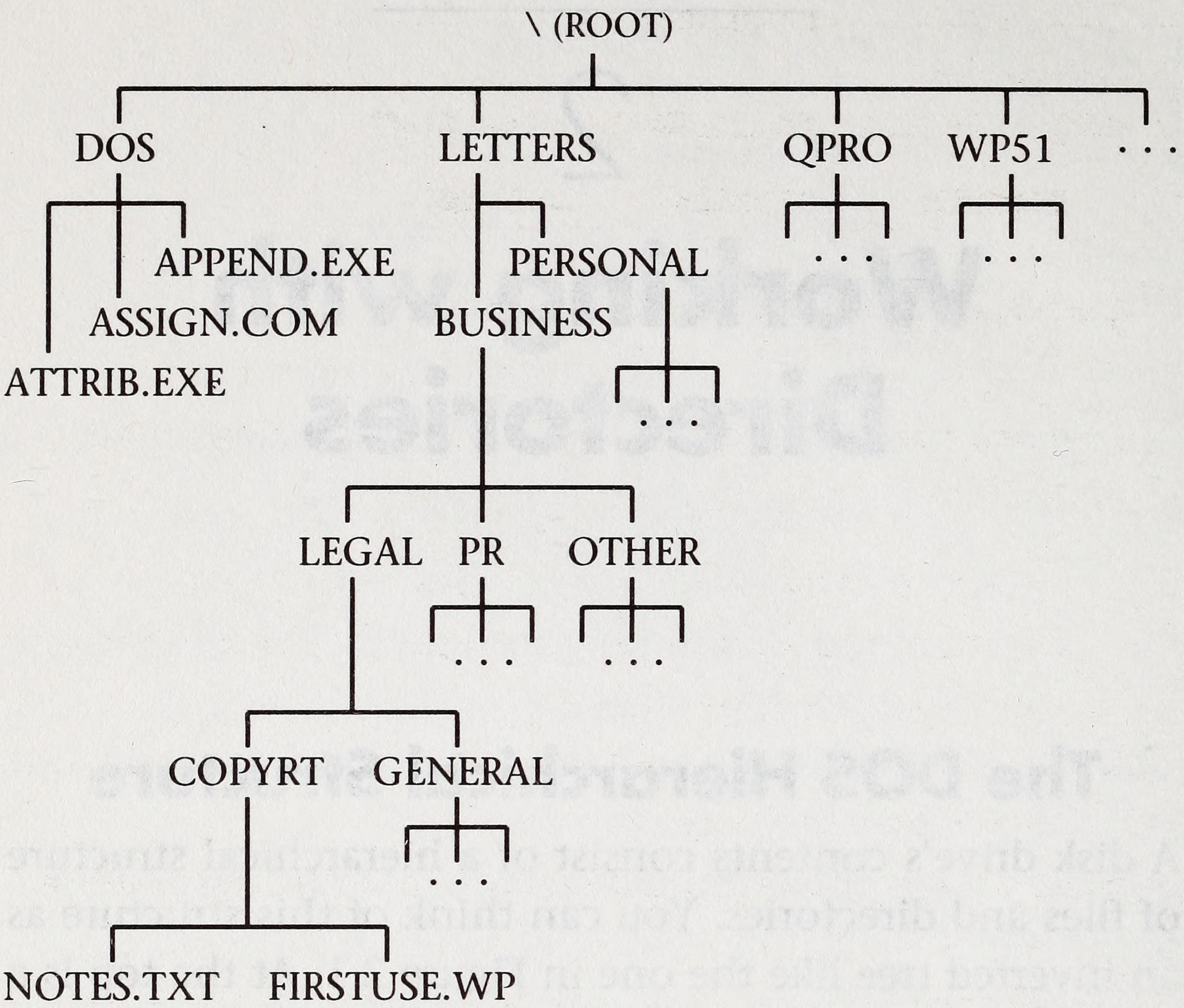


Figure 2.1 A typical directory structure

file or subdirectory is called the *parent directory* of that file or subdirectory. In Figure 2.1, \ is the parent directory of several directories, including DOS, LETTERS, QPRO, and WP51. Notice that each directory name in Figure 2.1 is a maximum of eight characters; this is a rule in DOS.

Changing Drives

When you first start the DOS Shell, it usually focuses its attention on drive C. You can tell that this is so because the drive C icon is highlighted in the drive icons area and the directory tree area shows its hierarchical structure. The drive whose contents you are viewing is known as the *current drive*.

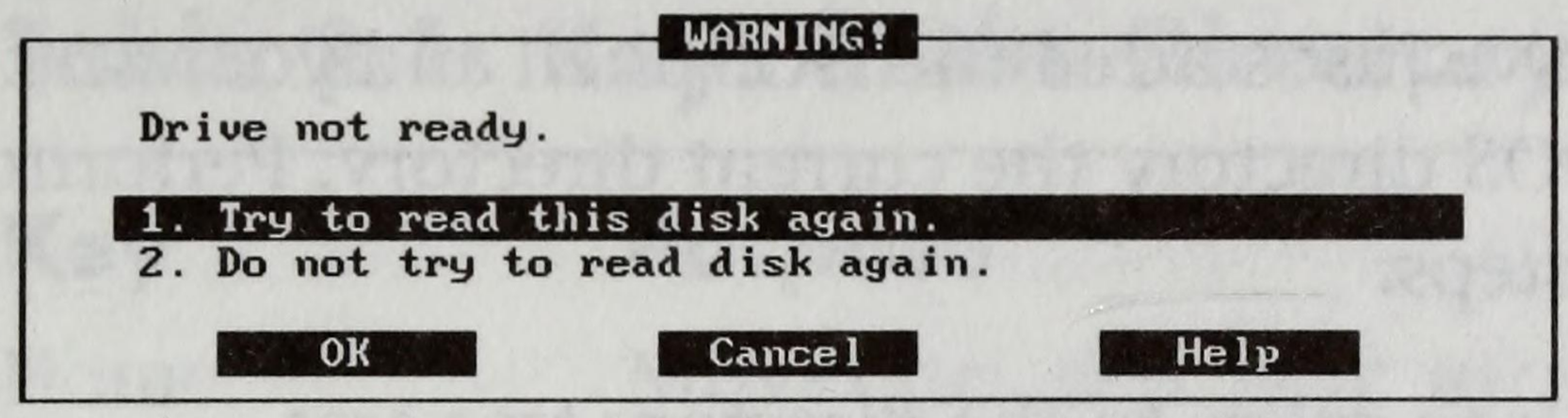


Figure 2.2 Drive warning message

Suppose you want to make drive A the current drive. From the DOS shell, here are the steps:

1. Insert a formatted floppy disk into drive A; any formatted disk will do. (See Lesson 14 to learn how to format a floppy disk.)
2. If necessary, keep pressing **Tab** until you move to the drive icons area.
3. Use → to highlight the **[A:]** icon and press **Enter**.

⇨ ***Note:*** If you're using the mouse, double-click on a drive's icon to make it the current drive.

If you don't insert a formatted floppy into drive A before selecting its icon, your PC will appear unresponsive for a few seconds and then show a warning message like the one in Figure 2.2, indicating that drive A is not ready. To clear the message, press **Esc** or insert a formatted disk into the drive and press **Enter**.

Changing Directories

The *current directory* is the directory you are in at any given time. For instance, when you first boot your PC, the current directory is the \ (root) directory. If you then move to one of your subdirectories, that subdirectory is the current directory.

Suppose you've just started the DOS Shell and you want to make the DOS directory the current directory. Perform the following steps:

1. Press **Tab** to move to the directory tree area.
2. Press ↓ repeatedly until the DOS directory is highlighted. (Use ↑ if you've gone too far.)

Tip

A quick way to navigate to a directory in the directory tree list is to press the first letter of its name. For example, pressing D in Figure 2.1 moves you to DOS, the first directory beginning with D. Pressing D again would move you to the next directory beginning with D if one existed.

Figure 2.3 shows how your screen appears after making the DOS directory the current directory. Table 2.1 shows the keys that you can use to move within the directory tree.

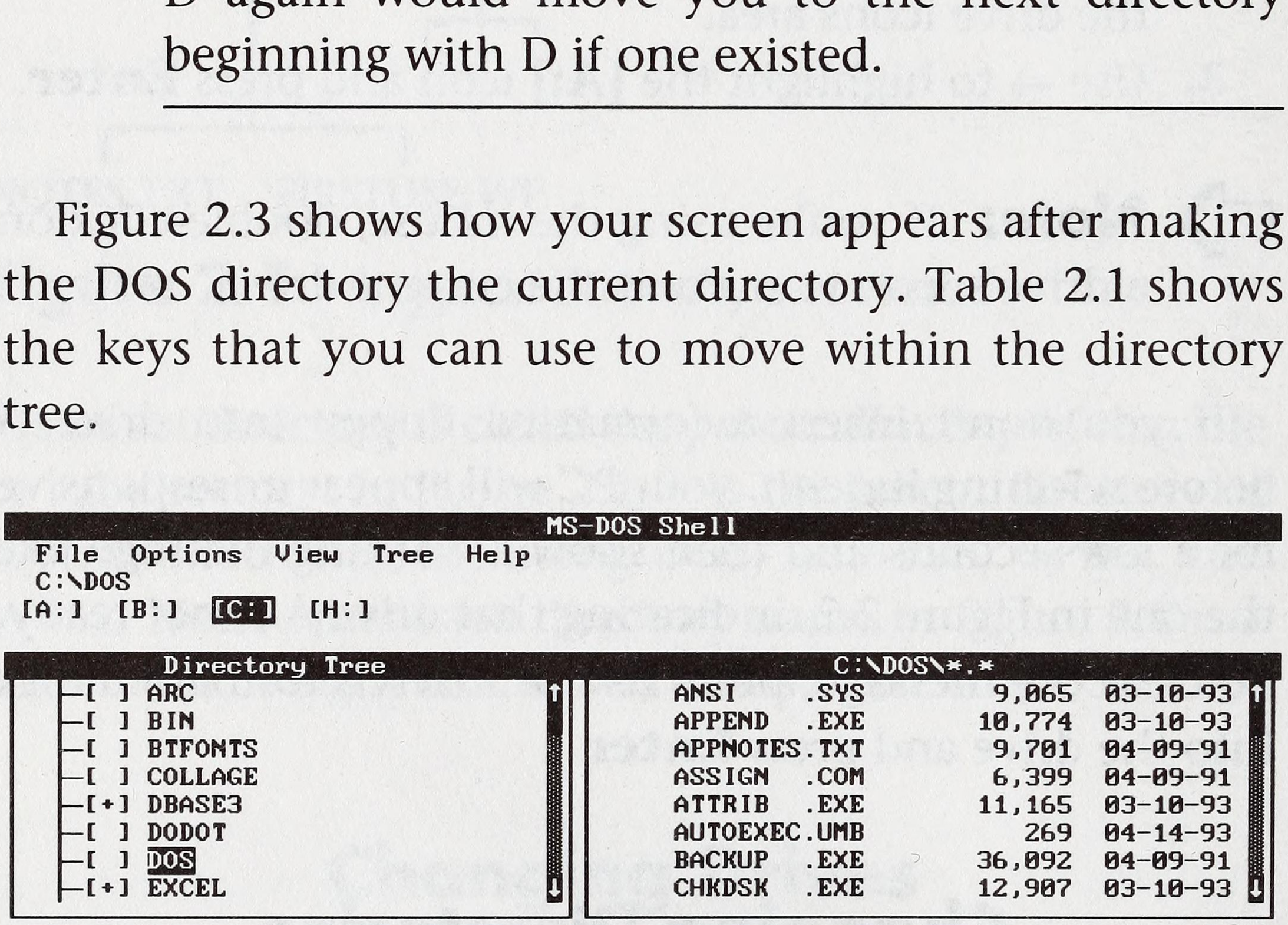


Figure 2.3 After making the DOS directory current

Table 2.1 Keys in the Directory Tree

Key	Purpose
Home	Moves to the first directory in the directory tree, which is the root directory of the current drive.
End	Moves to the last directory in the directory tree.
PgDn	Moves down by approximately 8 directories.
PgUp	Moves up by approximately 8 directories.

Notice that as you move among the directories, the file-list area changes its contents to show the files in the highlighted directory. Again, the directory whose contents you are viewing at any given time is the current directory.

With a mouse, use the scroll bar at the right of the directory tree to see other directories, as shown in Figure 2.4. (If you're new to the mouse, *dragging* is the term used to describe the procedure of holding down the left mouse button as you move the mouse, and then releasing the mouse button.) Click on an arrow to scroll one directory

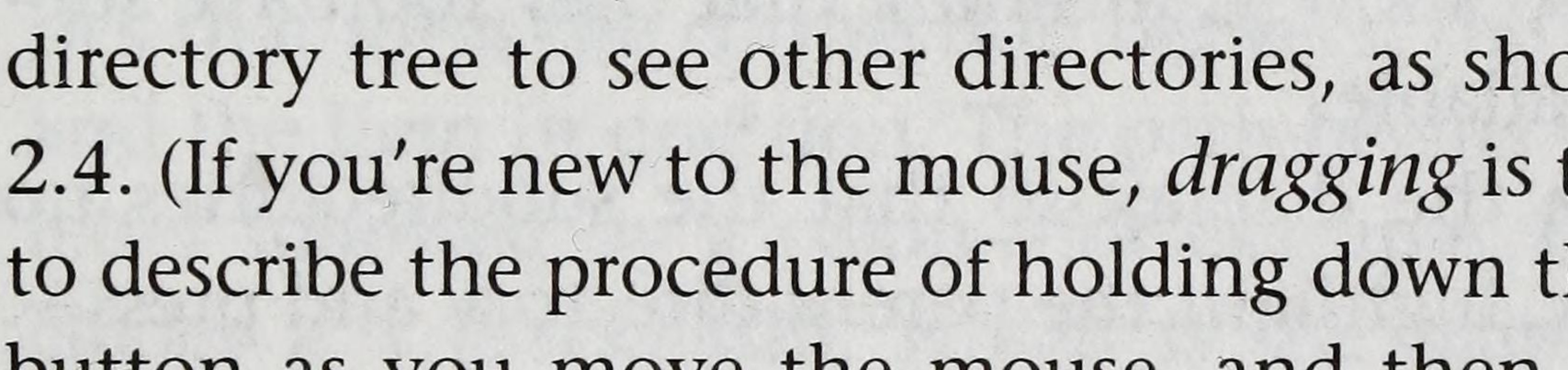


Figure 2.4 Clicking the scroll bar

at a time; to scroll continuously, point to an arrow and hold down the mouse button continuously. To move in large increments to a different area, drag the slider box up or down the scroll bar. Click above or below the slider box to scroll approximately eight directories at a time.

Revealing and Hiding Subdirectories

When a directory has a + preceding it in the directory tree, it means that it has subdirectories available. For example, if you want to see the subdirectories available from the EXCEL directory in Figure 2.3, use the following steps for revealing the subdirectories from the keyboard:

1. Press **e** or ↓ until you highlight the **EXCEL** directory in the directory tree.
2. Press **+**.

Figure 2.5 shows the effect on the directory-tree and file-list areas. Notice that the EXCELCBT and XLSTART directories also show +, meaning that they too have subdirectories available.

To collapse the display so that the subdirectories no longer appear, highlight the parent directory and press **–**. If you do this with the EXCEL directory, for example, the display reverts back to Figure 2.3.

Tip

The quickest way to reveal all available subdirectories on the current drive is to press Ctrl+* (asterisk) from anywhere within the directory tree.

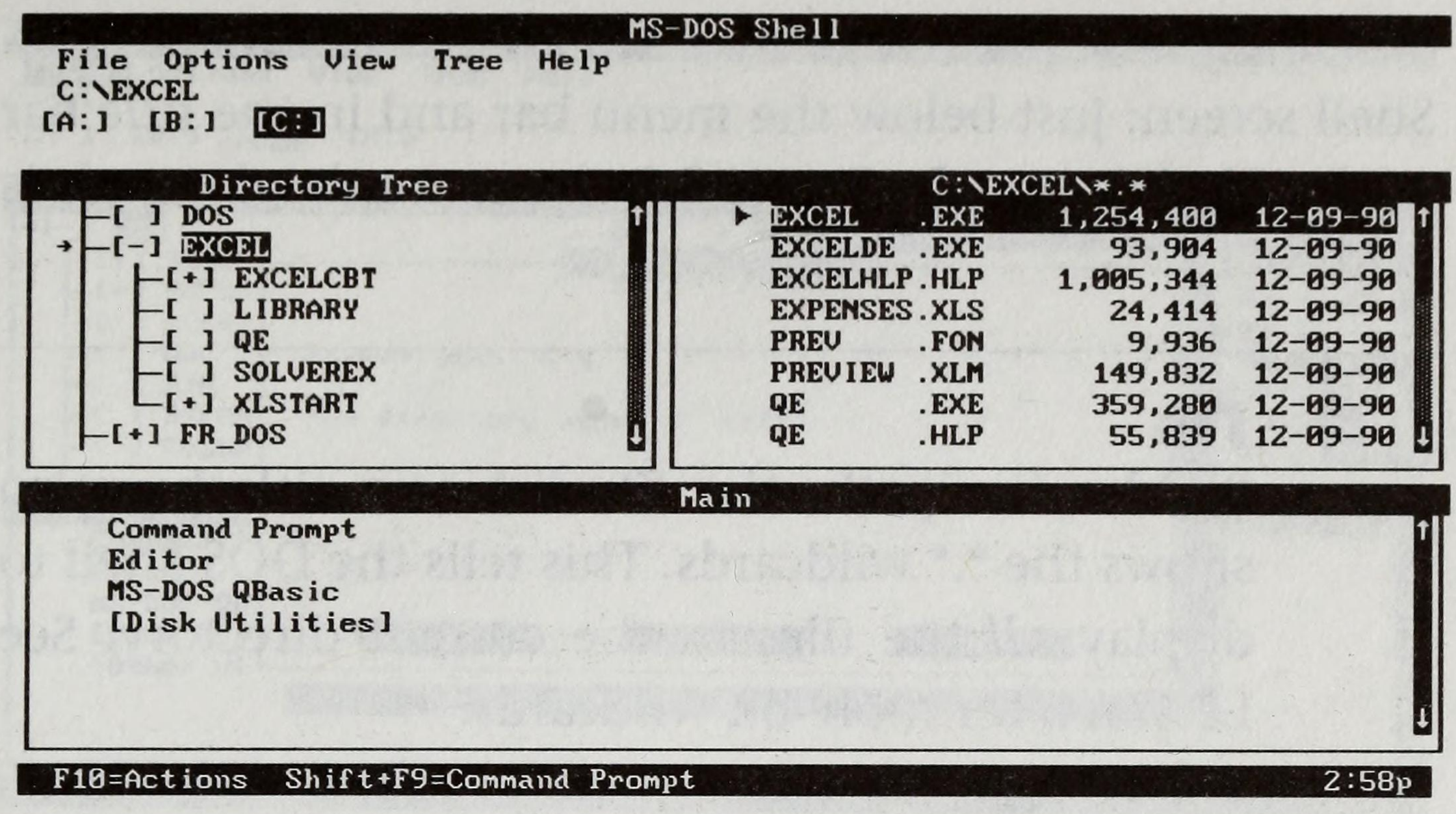


Figure 2.5 After pressing + to view subdirectories

If you're using the mouse, click on the + preceding a parent directory to show its subdirectories; to hide subdirectories, click on the – preceding the parent.

Paths

The *path* of a file or directory is the list of all the directories that form the connection between the root directory and the item in question. The path begins with the drive letter followed by a colon—for example, C:. It then includes a \ (backslash), which is the name of the root directory. A backslash also separates the names of the directories in a path. For example, here is the path for the file NOTES.TXT shown in Figure 2.1.

```
C:\LETTERS\BUSINESS\LEGAL\COPYRT
```

To show the path and filename together, use

```
C:\LETTERS\BUSINESS\LEGAL\COPYRT\NOTES.TXT
```

The current path is shown in two places on the DOS Shell screen: just below the menu bar and in the title bar of the file-list area. In Figure 2.5, for example, the path is C:\EXCEL.

Tip

Besides the path, the file list area title bar also shows the *.* wildcards. This tells the DOS Shell to display *all* the files in the current directory. See Lesson 8 for more on wildcards.

Creating a Directory

To create a new directory off the current directory, you use the File Create Directory command. Try the following example to create a directory named TEST off the root directory:

1. Press **Tab** until you move to the directory tree and press **Home** to move to the root directory of the current drive, C:\.
2. Select the **File Create Directory** command. The dialog box in Figure 2.6 appears.
3. Type **test** and press **Enter**. DOS creates the new TEST directory off of the root directory.
4. To see TEST in the directory tree, press **PgDn** (or use ↓ or T repeatedly) to bring it into view.

Removing a Directory

To remove a directory, you use the File Delete command (or press Del, its shortcut). For example, here's how to delete the TEST directory you created in the previous example from the DOS Shell:

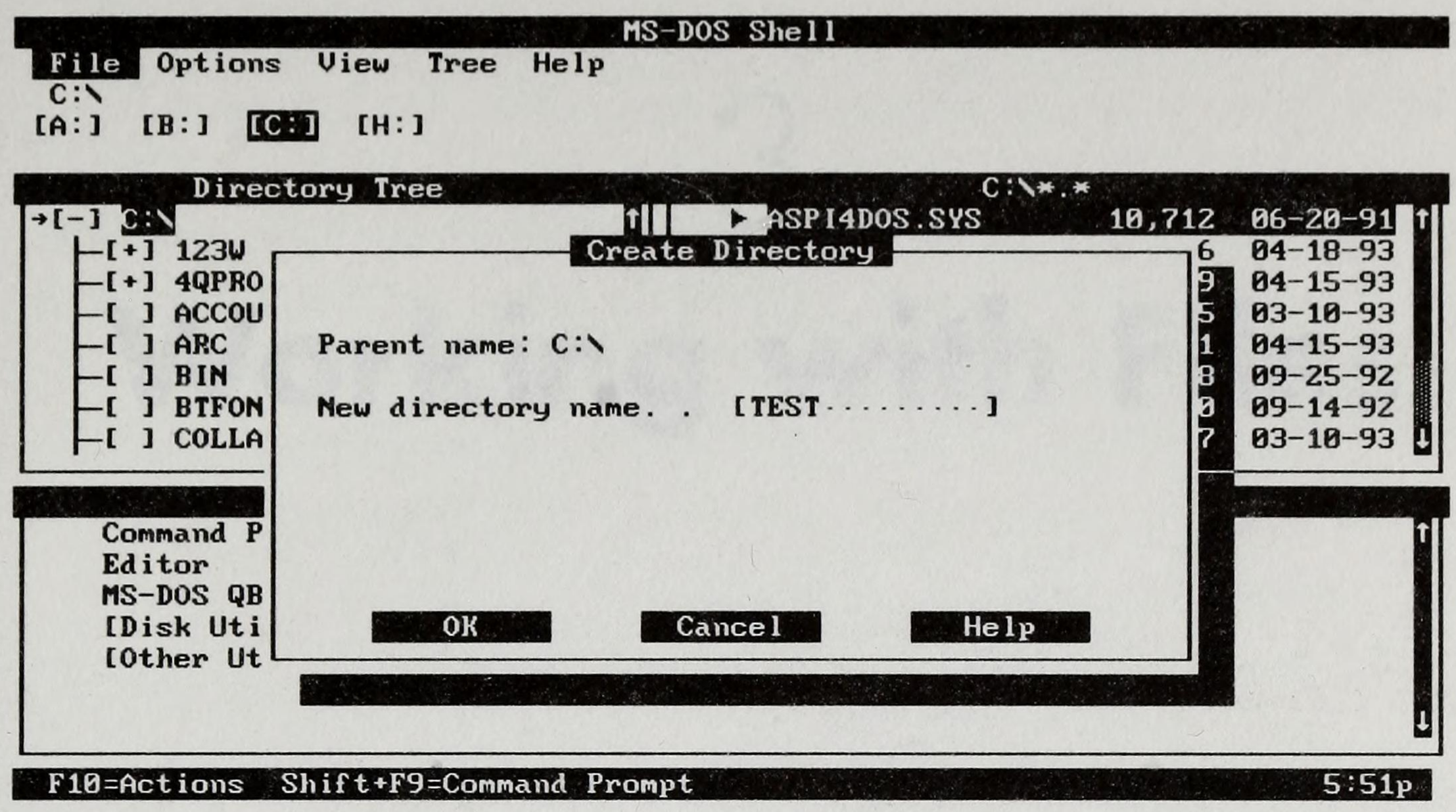


Figure 2.6 The File Create Directory dialog box

If you're working in the DOS Shell, a directory must be empty before you can remove it. Therefore, if a directory contains files or subdirectories, you'll need to delete them before you can remove the directory.

1. Press **t** or ↓ until you highlight the **TEST** directory in the directory tree.
2. Press **Del**. You'll then see a dialog box asking you to confirm the deletion.
3. Select the **Yes** option by pressing **Enter**.

Tip

In DOS 6, you can use the Deltree command at the MS-DOS command prompt to simultaneously delete a directory as well as all files and subdirectories within it. See Lesson 18.

◆ *Lesson* ◆

3

Working with Files

Naming Files

Filenames can be up to eight characters long with a three-character extension. If you attempt to use a name that is longer than eight characters, DOS disregards the extra characters. Here are some additional rules that apply to filenames:

- You can use any combination of letters, numbers, underscores (_), and hyphens (-). However, you cannot use the *, ?, |, /, <, or > characters because these have special meaning in DOS. In addition, you can use a period only once in a filename, to separate the filename from its extension.
- You cannot use spaces.
- You can use either uppercase or lowercase letters.

The following are all acceptable filenames:

CASH-93.WQ1 75%_PLAN.TX MKT&SLS.WP _BACKUP.BAK

$60PRICE.BAS 07-23-93.XLS 4403.DRV #D11.C

Although file extensions are optional, most veteran DOS users take advantage of them for these reasons:

- They make it easier to distinguish files from directories. (Directories share all the same naming conventions as files, but are rarely given extensions.)
- They help identify the type of file you are dealing with—for example, .TXT usually means a text file and .BAK a backup file.
- Most application programs automatically assign their own unique file extensions—for example, .XLS for Excel worksheet files and .WQ1 for Quattro Pro files.
- Using the same file extension for files that have a similar purpose lets you match them all by a single pattern and a wildcard character. For example, if you have several word processing files concerning legal action you have pending against widows and orphans, you might want to give them a .w&o extension, just to indicate to yourself what these files are about. Then, whenever you want to do some operation on these files, you could refer to them as *.w&o. (Lesson 8 will tell you all about wildcard characters.)

Copying Files

The File Copy command lets you copy a file to a different directory using the same filename or to the current directory under another filename. It's handy when you want to have a file available in more than one directory and for making backup copies of important files. You can also copy a file using drag-and-drop.

Copying a File Within a Directory

To copy a file within a directory, you must give it a different filename. For example, suppose you want to make a copy of the file README.TXT in your DOS directory and name it TEST.TXT in that same directory. From the DOS shell, here are the steps:

1. Press ↓ or **d** repeatedly until you select the **DOS** directory in the directory tree. All its files will appear in the file list.
2. Press **Tab** to move to the file list. Then press ↓ or **r** repeatedly until README.TXT is highlighted.
3. Select the **File Copy** command or press its shortcut, **F8**. The dialog box shown in Figure 3.1 appears.
4. Type **test.txt** for the To: location and press **Enter**.

DOS makes a copy of the file and names it TEST.TXT. To see the new file in the alphabetized file list, press **PgDn**

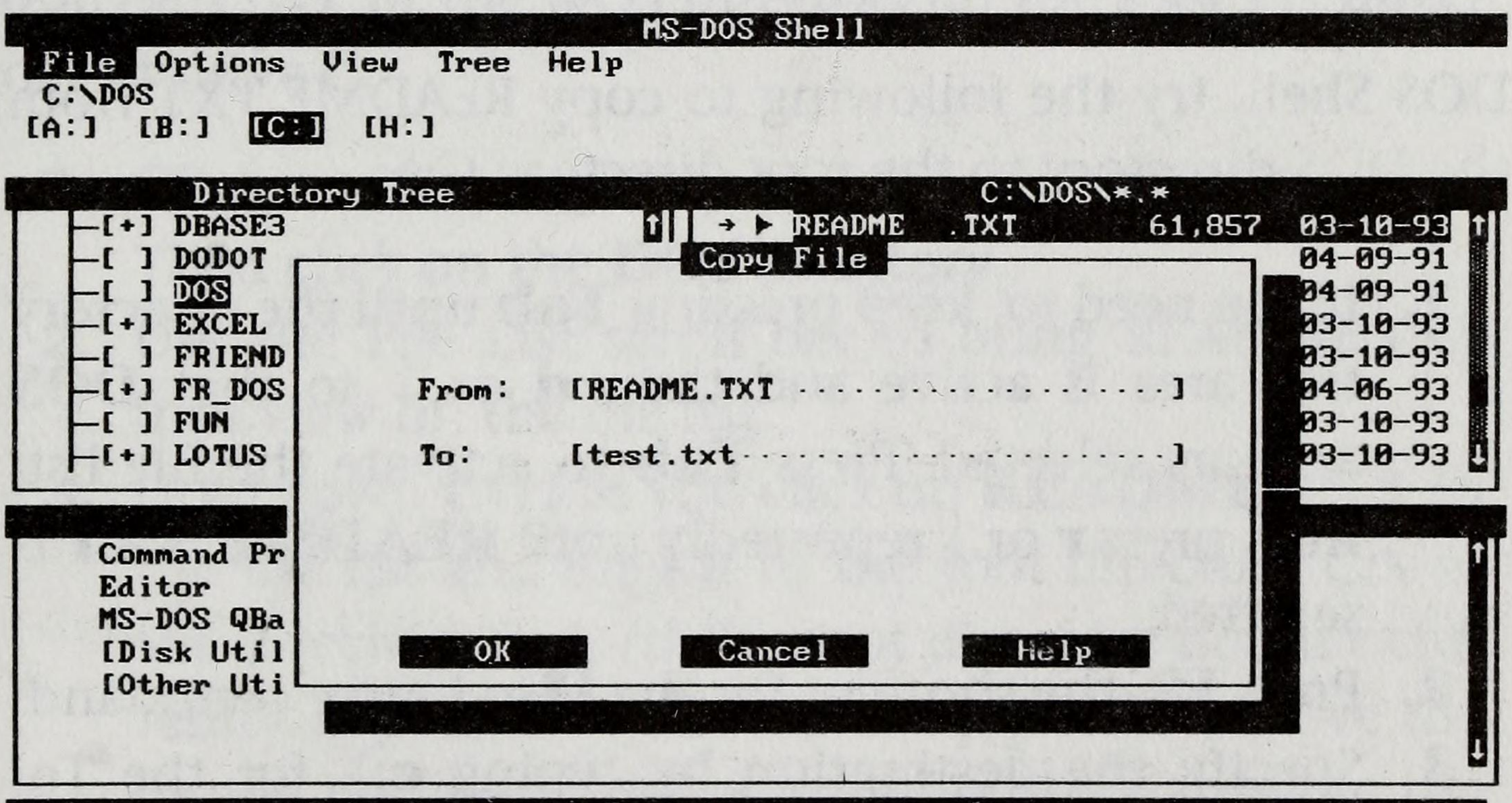


Figure 3.1 The File Copy File dialog box

until it comes into view. (In Lesson 4, you'll learn how to delete this file.)

Tip

A quick way to select a file in the alphabetized file list is to press the first letter of its name. For example, to select README.TXT when viewing the contents of the DOS directory, simply press *r*. This selects the first file in the list that begins with that letter, which in DOS 6 is RAMDRIVE.SYS. Pressing *r* again moves to the second file in the list that begins with *R*, which in this case is the desired file, README.TXT.

Copying a File to Another Directory

By specifying a To: path in the File Copy command, you can copy a file to another directory. For example, from the DOS Shell, try the following to copy README.TXT from the DOS directory to the root directory C:\:

1. If you need to, keep pressing **Tab** until the directory tree area is active and then **d** or ↓ so that **DOS** appears selected. Press **Tab** to activate the file list, then press **r** or ↓ repeatedly until **README.TXT** is selected.
2. Press **F8**, the shortcut for the **File Copy** command.
3. Specify the destination by typing **c:** for the To: location. Press **Enter** to confirm.

Tip

When copying files to different directories on the current drive, the drive portion of the path is optional. For example, you could use \ in step 3 of the previous procedure instead of c:\test.txt and still get the same results. See Lesson 22 for more on these and other command line shortcuts.

USING DRAG-AND-DROP TO COPY FILES TO ANOTHER DIRECTORY

From the DOS Shell, you can copy a file to another directory under the same filename by using the mouse to *drag-and-drop*—holding down the Ctrl key while dragging a filename. In fact, by selecting more than one file (see Lesson 6) you can use drag-and-drop to copy multiple files simultaneously.

From the DOS Shell, try this process to copy README.TXT in the DOS directory to the root (\) directory of drive C:

1. Click on the directory tree area to make it active. Then click on the **DOS** directory.
2. Use the File List scroll bar to bring README.TXT into view on the file list.
3. Hold down **Ctrl** as you click on **README.TXT** in the file list and drag it to the root directory C:\ in the directory tree. (If the root directory doesn't currently appear with the directory tree list, move the mouse pointer to the upward pointing arrow in the directory list scroll bar to scroll the root directory

into view; be sure to keep both the Ctrl key and the mouse button held down the entire time.) When the mouse pointer is at the root directory, you'll see the following message at the foot of the screen:

```
Copy README.TXT to \
```

4. With the mouse pointer located on the root directory C:\, release the mouse button. Figure 3.2 shows the dialog box that appears asking you to confirm the operation.
5. Click on the **Yes** button.

Tip

When copying or moving files using drag-and-drop, you can save considerable time and trouble by using the Dual File Lists view, which presents two directory trees and file lists at the same time. See Lesson 7.

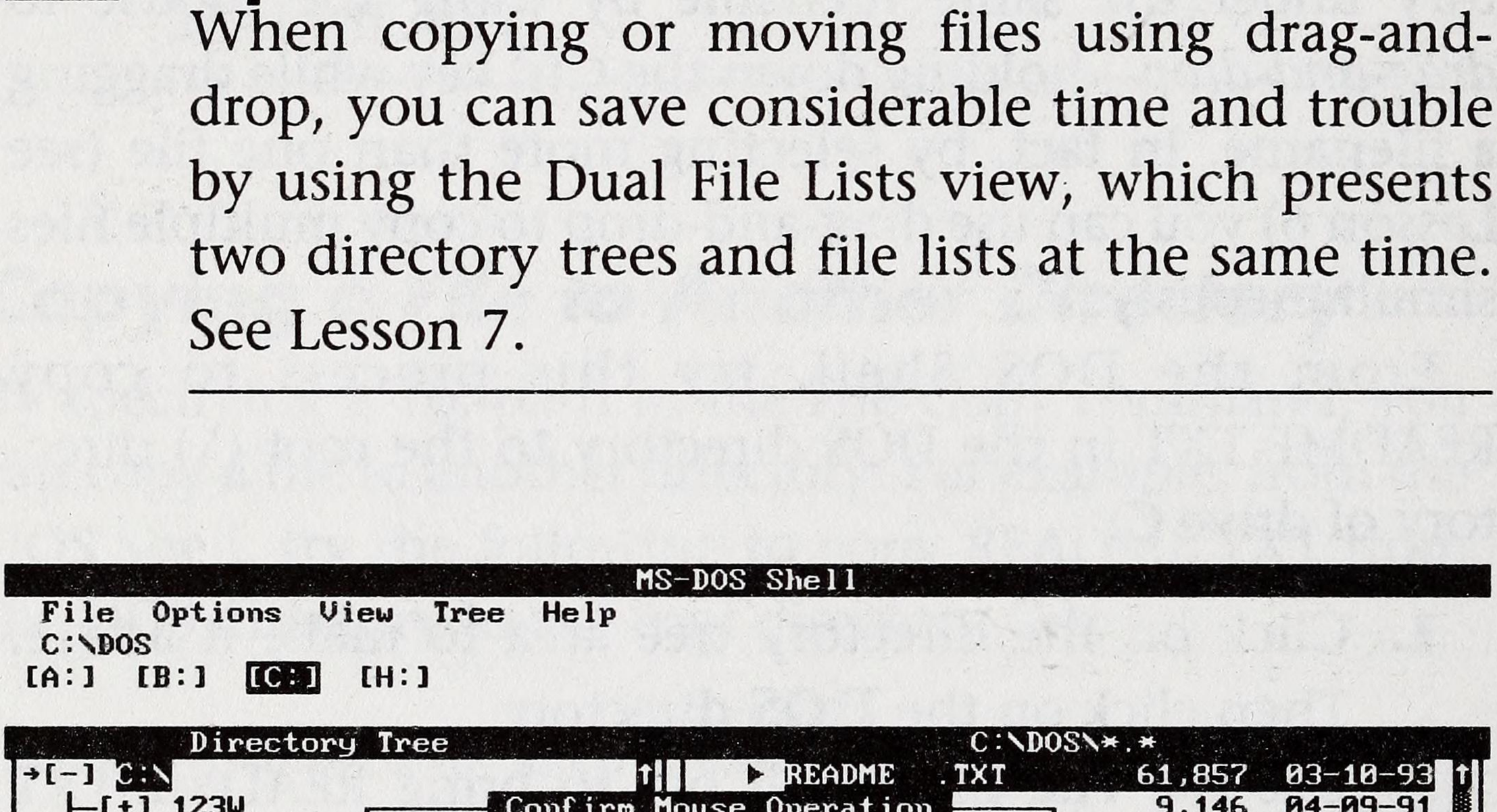


Figure 3.2 Confirmation when copying a file

Copying Files between Drives

To copy a file from one drive to another using the keyboard, you must specify the target drive and path for the File Copy command's To: location. From the DOS Shell, here are the steps for copying README.TXT in C:\DOS to the root directory of drive A:

1. Insert a formatted floppy disk in drive A. (To format a floppy, see Lesson 14.)
2. Press **Tab** until the directory tree area is active. Then press ↓ or **d** until you select the **DOS** directory.
3. Press **Tab** to move to the file list area. Then press ↓ or **r** until you select **README.TXT** in this list.
4. Choose the **File Copy** command or press **F8**.
5. Type **a:** for the To: location and press **Enter**.

To see the copy of README.TXT on drive A, move to the drive-icons area and make drive A the current drive.

USING DRAG-AND-DROP TO COPY FILES BETWEEN DRIVES

From the DOS Shell, you can use the mouse to copy a file to another file using drag-and-drop. For example, here's how you can drag-and-drop README.TXT in C:\DOS to the root directory of drive A:

1. Insert a formatted floppy disk in drive A. (To format a floppy, see Lesson 14.)
2. Click on the directory tree area to make it active. Then click on the **DOS** directory.
3. Use the file list scroll bar so that README.TXT appears in the file list.

4. Click on **README.TXT** in the file list and drag it to the **drive A** icon in the drive-icon area above. (Holding down the Ctrl key isn't necessary when copying files between drives.)
5. Release the mouse button.
6. Select **Yes** (press **Enter**) in response to the confirmation message.

◆ *Lesson* ◆

4

More File Operations

Moving Files

To move a file using the keyboard, you use the File Move command or its shortcut, F7. For example, suppose you want to move the file OS2.TXT from the DOS directory to the root directory of drive C. From the DOS Shell, follow these steps:

⇨ ***Note:*** OS2.TXT only exists in DOS 6; if you're working with DOS 5, try this example using APPNOTES.TXT instead.

1. Press **Tab** until the directory tree area is active. Then press ↓ or **d** until you select the **DOS** directory.
2. Press **Tab** to move to the file list area. Then press ↓ or **o** until you select **OS2.TXT** in this list.
3. Choose the **File Move** command or press its shortcut, **F7**. Figure 4.1 shows the dialog box that appears. Notice that the selected file OS2.TXT is already included in the From: text box.

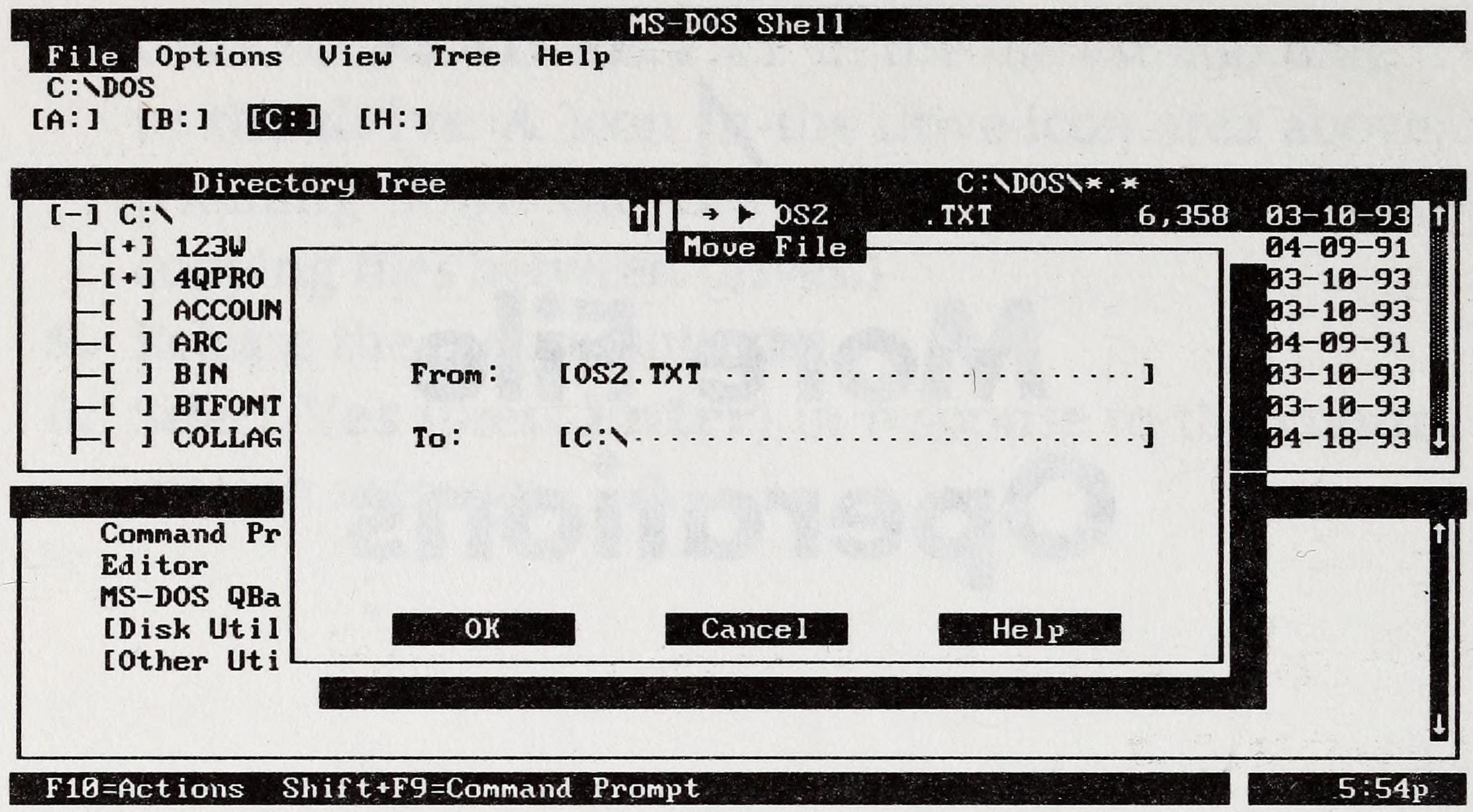


Figure 4.1 The File Move File dialog box

4. Type **c:** for the To: location and press **Enter**. OS2.TXT is removed from C:\DOS and now appears in C:\.
5. To see the file in its new location, keep pressing **Tab** until you return to the directory list area. Then press **Home** to move to the root **C:** in the directory list.
6. To return OS2.TXT to its original location in C:\DOS, press **Tab** to move the file list area. Then press ↓ or **o** until **OS2.TXT** is selected. Choose the **File Move** command, type **c:\dos** for the To: location and press **Enter**.

Tip

DOS 6 includes the new Move command which allows you to move files at the MS-DOS command prompt. See Lesson 20.

Moving Files to Another Drive

You can also use File Move to move a file to another drive. Just enter the new drive and target path in the To: location. For example, if you used d:\backup for the To: location in the previous procedure, DOS would copy OS2.TXT to the BACKUP directory on drive D.

Moving Files by Dragging and Dropping

From the DOS Shell, you can move a file to another directory by using the mouse to drag-and-drop. You use the same technique as dragging and dropping to copy a file (see Lesson 3) with these exceptions:

- To move a file to another drive, hold down **Alt** as you click on the filename in the file list area and drag it to the destination drive icon in the drive icon area. (If you don't use Alt, the DOS Shell assumes you want to copy the file instead.)
- To move a file to another directory on the same drive, simply click on the filename in the file list area and drag it to the destination directory in the directory tree area.

Tip

Just as when you're copying files, using the Dual File Lists view is handy when moving files because it lets you see the contents of two directories at the same time. See Lesson 7.

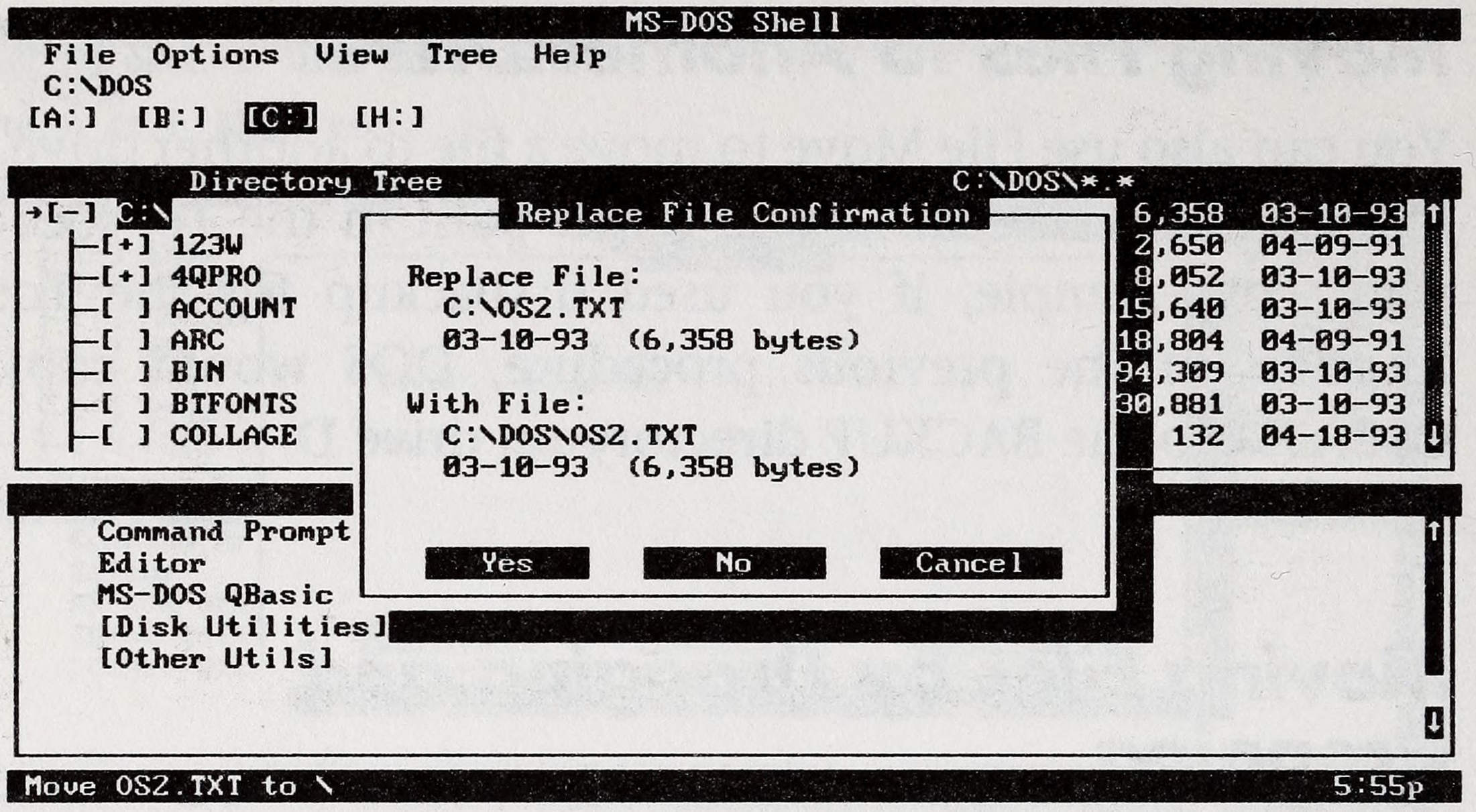


Figure 4.2 The dialog box when overwriting an existing file while moving

Duplicate Filenames

If, during a move, a duplicate filename already exists in the To: location, DOS automatically warns you through a dialog box like the one in Figure 4.2. For instance, imagine that when you were moving OS2.TXT to the root directory of drive C, a file with that same name already existed in that location. You'd then see a confirmation dialog box like the one in Figure 4.2 asking whether you want to overwrite the existing file with the one you're moving. At this point you must either select Yes, which replaces the file; or No or Cancel, both of which leave the original file intact and abandon the move operation.

Renaming Files

Renaming a File As You Move It

When you use the File Move command to move a file into a different directory, the file you move will have the same

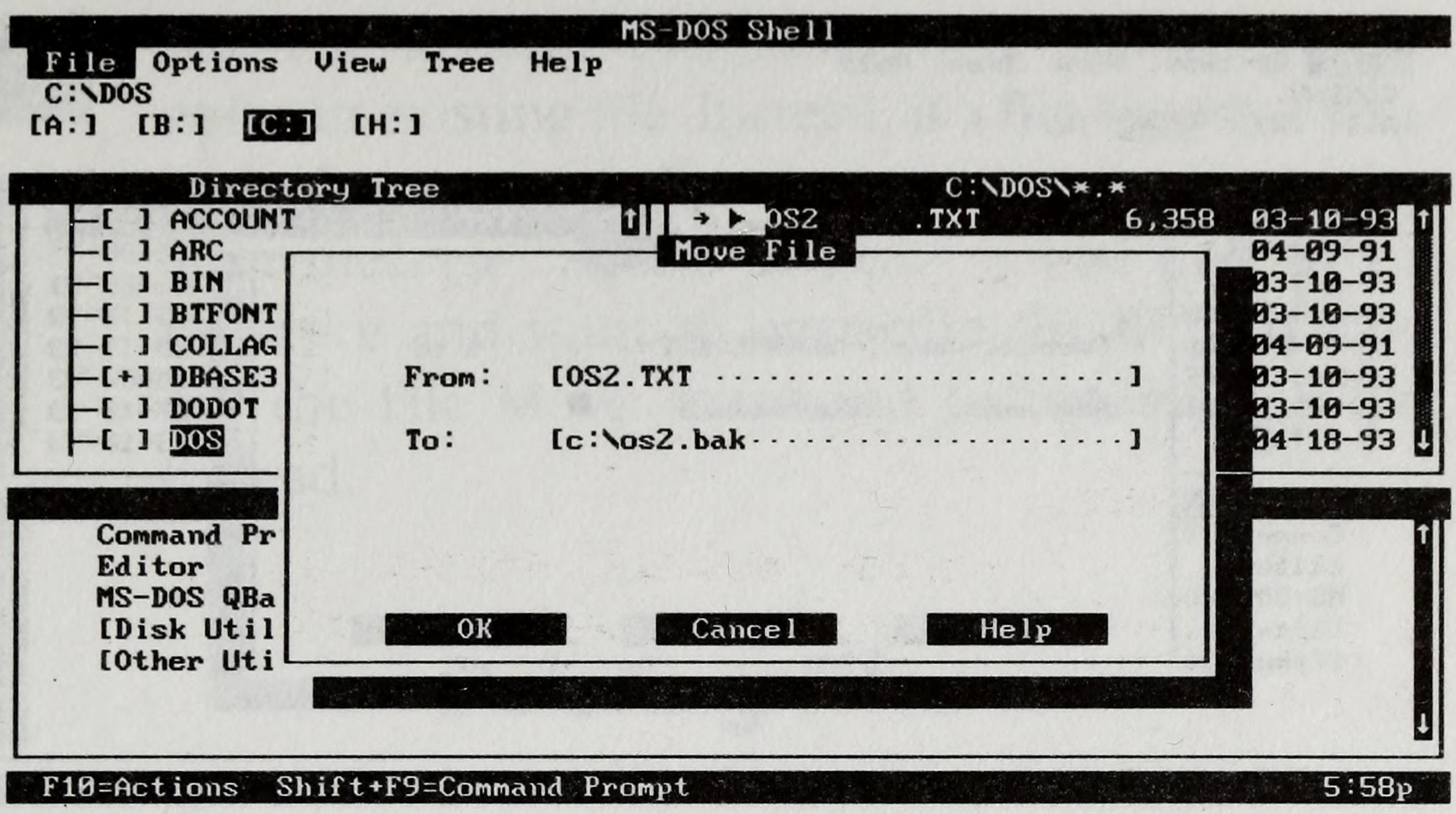


Figure 4.3 Renaming OS2.TXT

name in the new directory, unless you specify otherwise in the To: location. Building on the previous example, suppose in the process of moving O2S.TXT from the DOS directory to the root directory of drive C, you wanted to call the file OS2.BAK. Figure 4.3 shows the settings you would use for the File Move command. Notice that the new name, OS2.BAK, is appended to the target path name after a backslash.

Renaming a File in Its Current Location

The File Rename command enables you to rename a file in its current location. For example, from the DOS Shell here's how to rename README.TXT in the DOS directory to DOSREAD.ME:

1. Press **Tab** until the directory tree area is active, then press ↓ or **d** until you select the **DOS** directory.

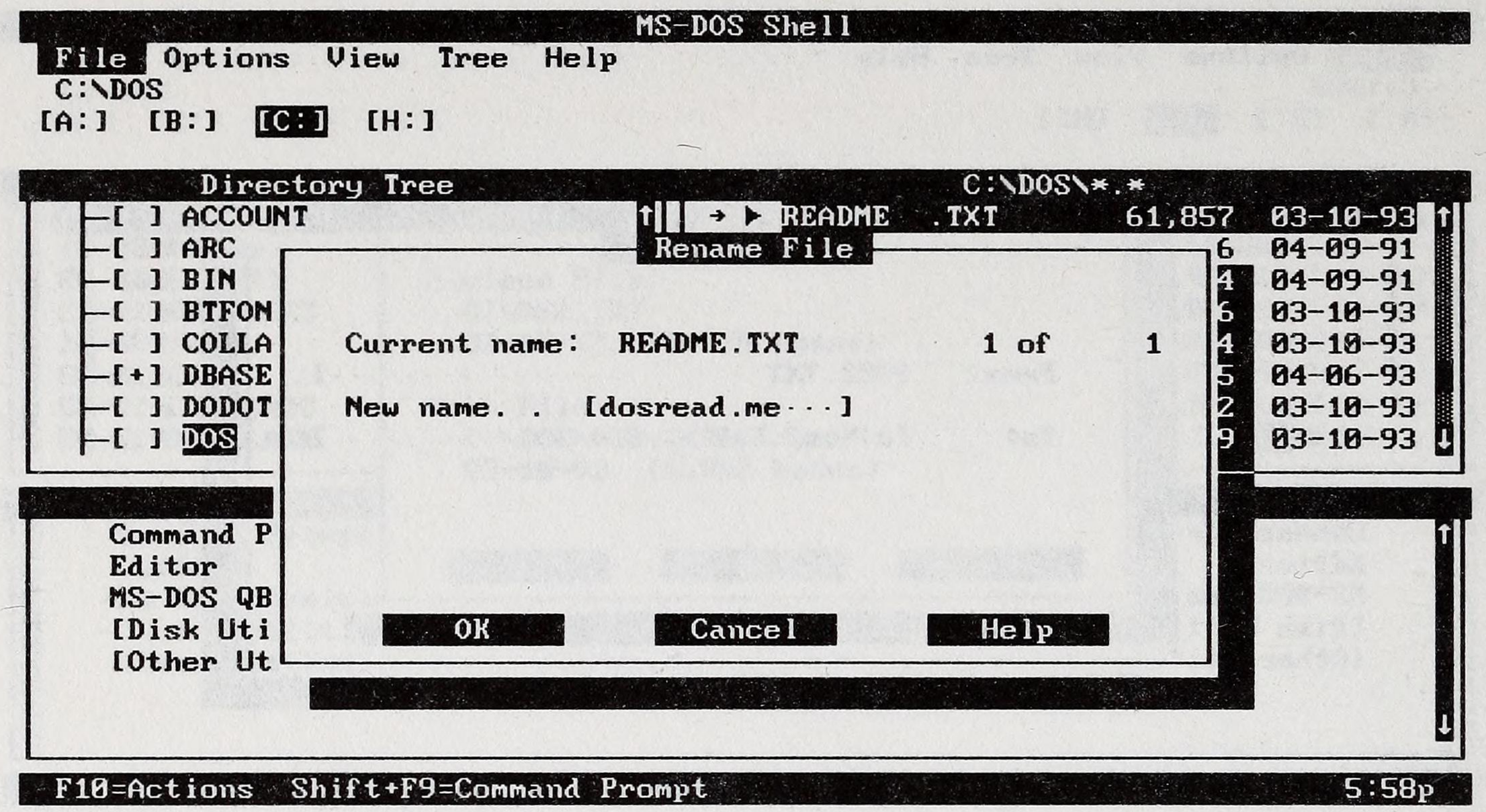


Figure 4.4 The File Rename File dialog box

Press **Tab** to move to the file list area, then press ↓ or **r** until you select **README.TXT**.

2. Choose the **File Rename** command. Figure 4.4 shows the dialog box that appears.
3. Type **DOSREAD.ME** for the new name and press **Enter**.
4. To view the newly named file, press ↑ or **d** until **DOSREAD.ME** is selected in the file list.
5. Change the file back to its original name by selecting the **File Rename** command again, typing **README.TXT** and then pressing **Enter**.

Although the name of the file changes (causing it to appear in a different location in the DOS Shell's alphabetized file list), the file itself remains in exactly the same location on the disk.

The File Rename command won't let you overwrite an existing file. Instead, if a file already exists with the new name you've chosen, you'll see the error message "Access denied." If you find this a nuisance and want to overwrite the file anyway, use the File Move command to rename the file instead.

Deleting Files

When you no longer need a file, you can remove it from disk using the File Delete command or its shortcut, Del. For example, if you followed along in Lesson 3, you created a copy of README.TXT in your DOS directory and named it TEST.TXT. Here's how to delete TEST.TXT from disk using the DOS Shell:

1. Press **Tab** until the directory tree area is active, then press ↓ or **d** until you select the **DOS** directory. Press **Tab** to move to the file list area, then press ↓ or **t** until you select **TEST.TXT**.
2. Press **Del**. A confirmation message appears.
3. Select **Yes** by pressing **Enter**.

Tip

You can also use Del to delete a directory, provided the directory is empty. And, if you've unintentionally deleted a file, you can recover it with the Undelete utility. See Lesson 23.

◆ *Lesson* ◆

5

Running Programs

Files that can be Run from DOS

By far the most common task you'll perform in DOS is running programs. As you become more acquainted with DOS, you'll find that the easiest way to run a program is from the DOS command line; in the meantime, you may find it simpler from the DOS Shell.

Only certain types of files can be run from DOS:

.EXE or .COM files These are executable program files that are stored in machine-language form. When you run one of these program files, DOS copies it into RAM and the microprocessor executes it.

.BAT files These are batch files, a special form of text file that DOS reads and interprets. Batch files are used primarily to run executable program files in suc-

.BAT files *(continued)*

cession—for example, a screen capture program followed by a spreadsheet program. But as you'll learn in Lesson 29, you can use them for much more than that.

The DOS Shell also has the unique ability to run two other types of files, which can't be run from the DOS command line.

.TXT files

These are ASCII (American Standard Code for Information Interchange) text files that are automatically loaded into the MS-DOS Editor for viewing and changing. (See Lesson 13.)

.BAS files

These are QBasic program files. DOS 6 comes with five programs written in QBasic (GORILLA.BAS, MONEY.BAS, NIBBLES.BAS, REMLINE.BAS, and WSCOLOR.BAS) that you can execute directly from the DOS Shell.

Tip

The only reason that you can "run" .TXT and .BAS files from the DOS Shell is that file associations have been created for them. You can create your own file associations using the File Associate command. See Lesson 11.

Running a Program from the File List

One way to run a program from the DOS Shell is to open its executable file. To get a feel for how to do this, try running the DOS Help program:

1. Press **Tab** until the **DOS** directory is selected in the directory tree. Then press **Tab** to move to the file list area.
2. In the file list, choose **HELP.COM** by pressing **h** or ↓ so that this file is highlighted. Then press **Enter**. (Another option is to use the File Open command and specify HELP.COM.) After a slight pause, you'll see the MS-DOS Help Command Reference shown in Figure 5.1 listing all the available commands.

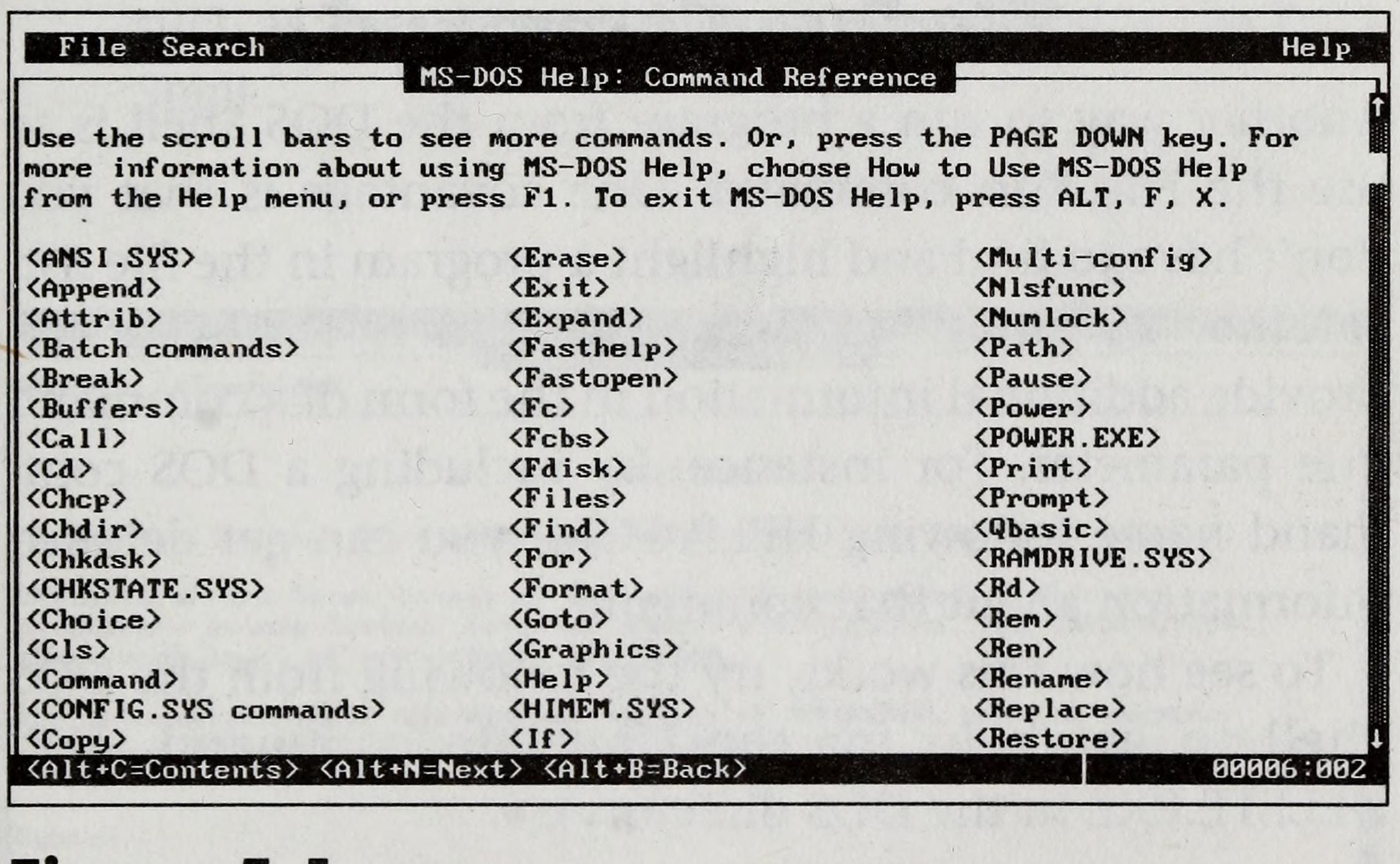


Figure 5.1 The initial HELP.COM screen

Tip

If you're using a mouse, you can run a program simply by double-clicking on its program file in the file list.

3. To view the entire Command Reference, press **PgDn** twice. To see information about a particular command, highlight the command and then press **Enter**.
4. To return to the DOS Shell, choose the **File Exit** command. You'll then be prompted to press any key such as **Esc**; do so and you'll be returned to the DOS Shell.

Running a Program with the File Run Command

Another way to run a program from the DOS Shell is to use the File Run command. One advantage is that you don't have to find and highlight a program in the file list. Another advantage of using File Run is that you can provide additional information in the form of a command line parameter. For instance, by including a DOS command name following HELP.COM, you can get detailed information about that command.

To see how this works, try the following from the DOS Shell to get help for the Undelete command (UNDELETE.EXE in the DOS directory):

1. Choose the **File Run** command.
2. Type **c:\dos\help undelete** just like in Figure 5.2 and press **Enter**. Figure 5.3 shows the screen that appears.

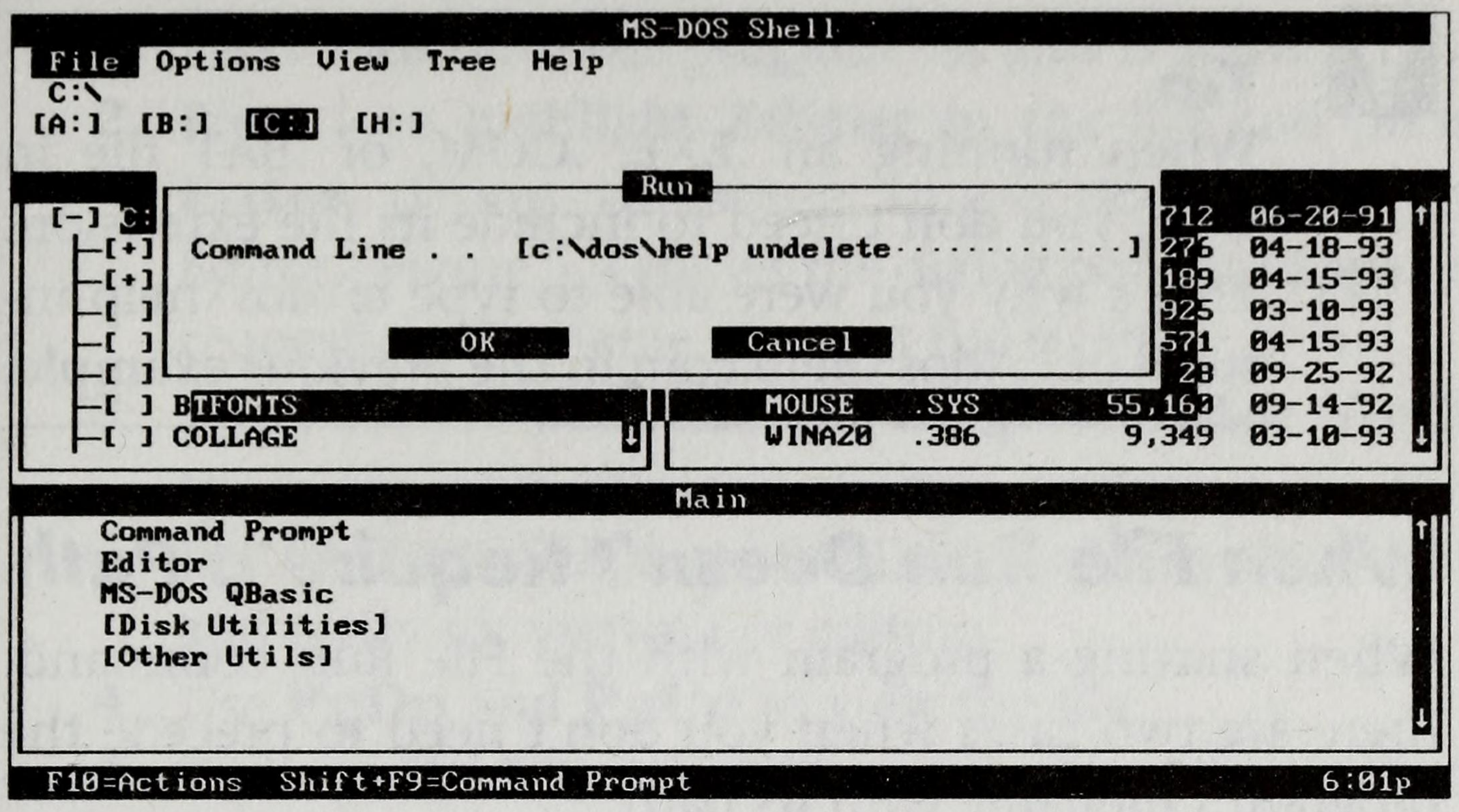


Figure 5.2 The File Run dialog box

3. To return to the DOS Shell, choose the **File Exit** command. You'll then be prompted to press any key such as **Esc**; do so and you'll be returned to the DOS Shell.

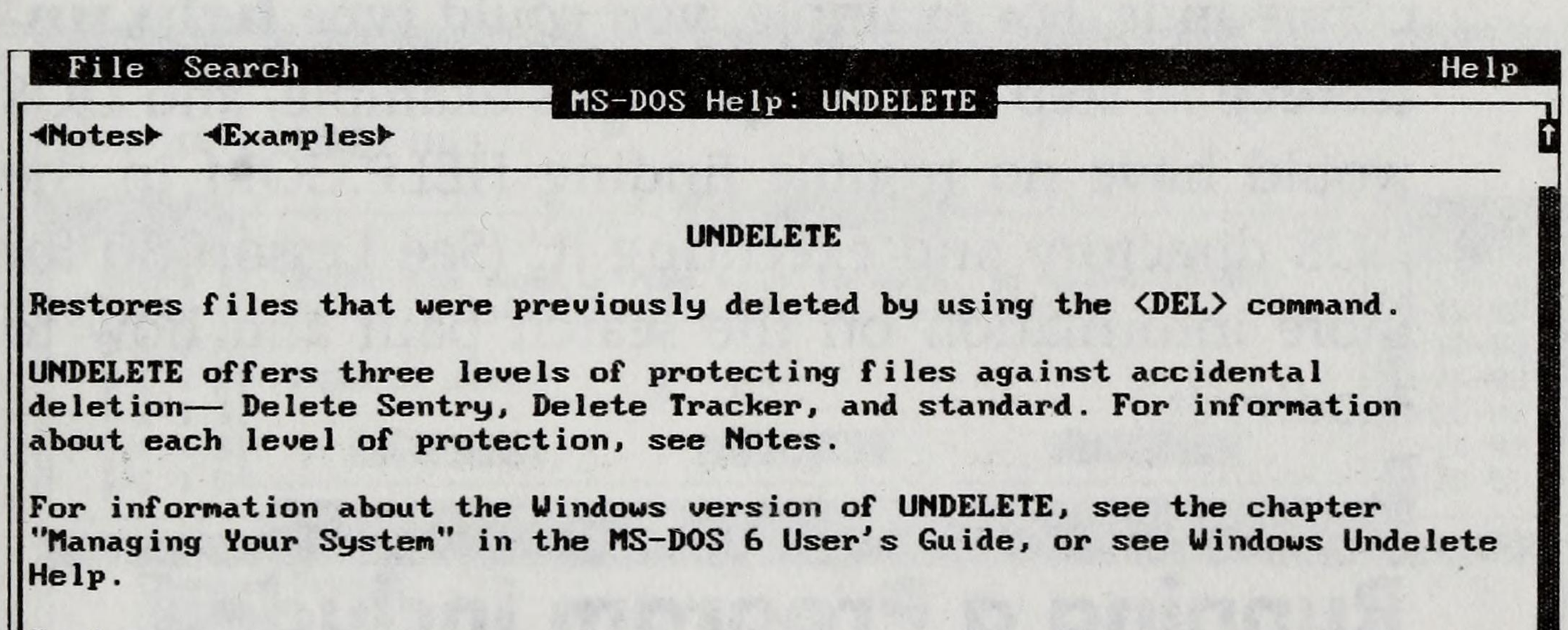


Figure 5.3 The Undelete command help screen

Tip

When running an .EXE, .COM, or .BAT file in DOS, you don't need to include its file extension. That's why you were able to type c:\dos\help instead of c:\dos\help.com in the previous example.

When File Run Doesn't Require a Path

When starting a program with the File Run command, there are two cases when you don't need to precede the program's filename with its path:

- When the program is in the current directory,
- When the program's directory is included in your search path. The *search path* is a list of directories that DOS searches when it can't find the program you're trying to run in the current directory. And because the DOS directory is almost always included in your search path, you don't usually need to provide any path information when running DOS commands. For example, you could type **help undelete** in step 2 of the previous example, and DOS would have no trouble finding HELP.COM in the DOS directory and executing it. (See Lesson 30 for more information on the search path and how to modify it.)

Running a Program Included in a Group

Another way to run a program from the DOS Shell is to select its name in the program list area. To run the MS-DOS Editor using the keyboard, for example:

1. Press **Tab** until the program list area is active.
2. Press ↓ to highlight **Editor** in the list and press **Enter**. (Using the mouse, simply double-click on Editor.) Figure 5.4 shows the dialog box that appears requesting the name of a text file to edit.
3. Type **c:\dos\readme.txt** and press **Enter**. You'll then see the MS-DOS Editor screen with the README.TXT file from the DOS directory displayed and ready for viewing or editing.
4. Use **PgDn** and **PgUp** to view the file.
5. To leave the MS-DOS Editor and return to the DOS Shell, select the **File Exit** command from the Editor menu.

Table 5.1 shows the names of the programs that are available in the program list area and their corresponding program files. All these program files are located in the DOS directory.

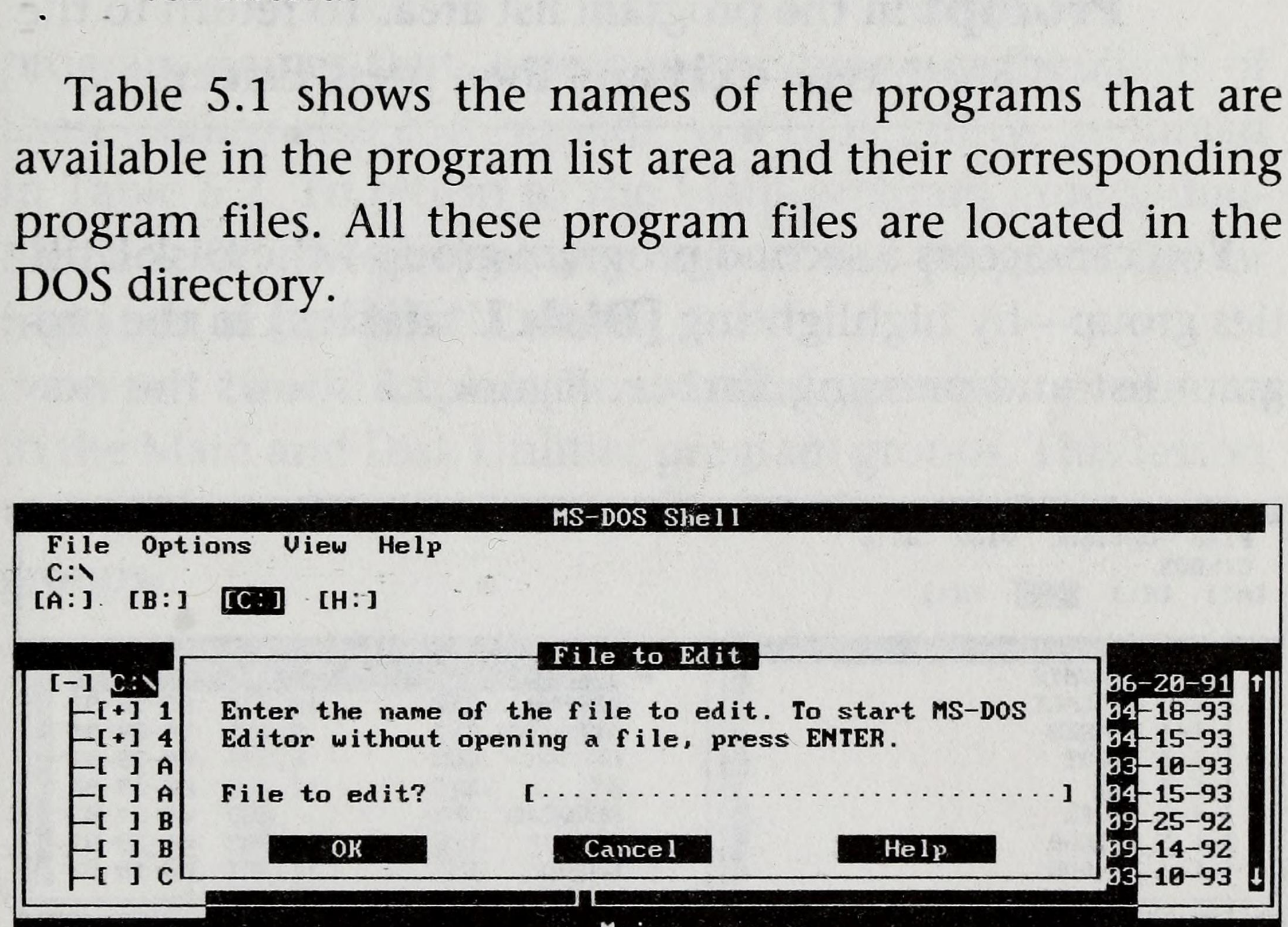


Figure 5.4 Running the MS-DOS Editor

Table 5.1 Main Program Group Programs

Name	Program File
Command Prompt	COMMAND.COM
Edit	EDIT.COM
MS-DOS QBasic	QBASIC.EXE

Tip

To temporarily leave the DOS Shell and return to the DOS command line, select **Command Prompt** in the program list area. To return to the DOS Shell, type **exit** and then press **Enter**.

You can access a second program group—the Disk Utilities group—by highlighting **[Disk Utilities]** in the program list and pressing **Enter**. Figure 5.5 shows the new

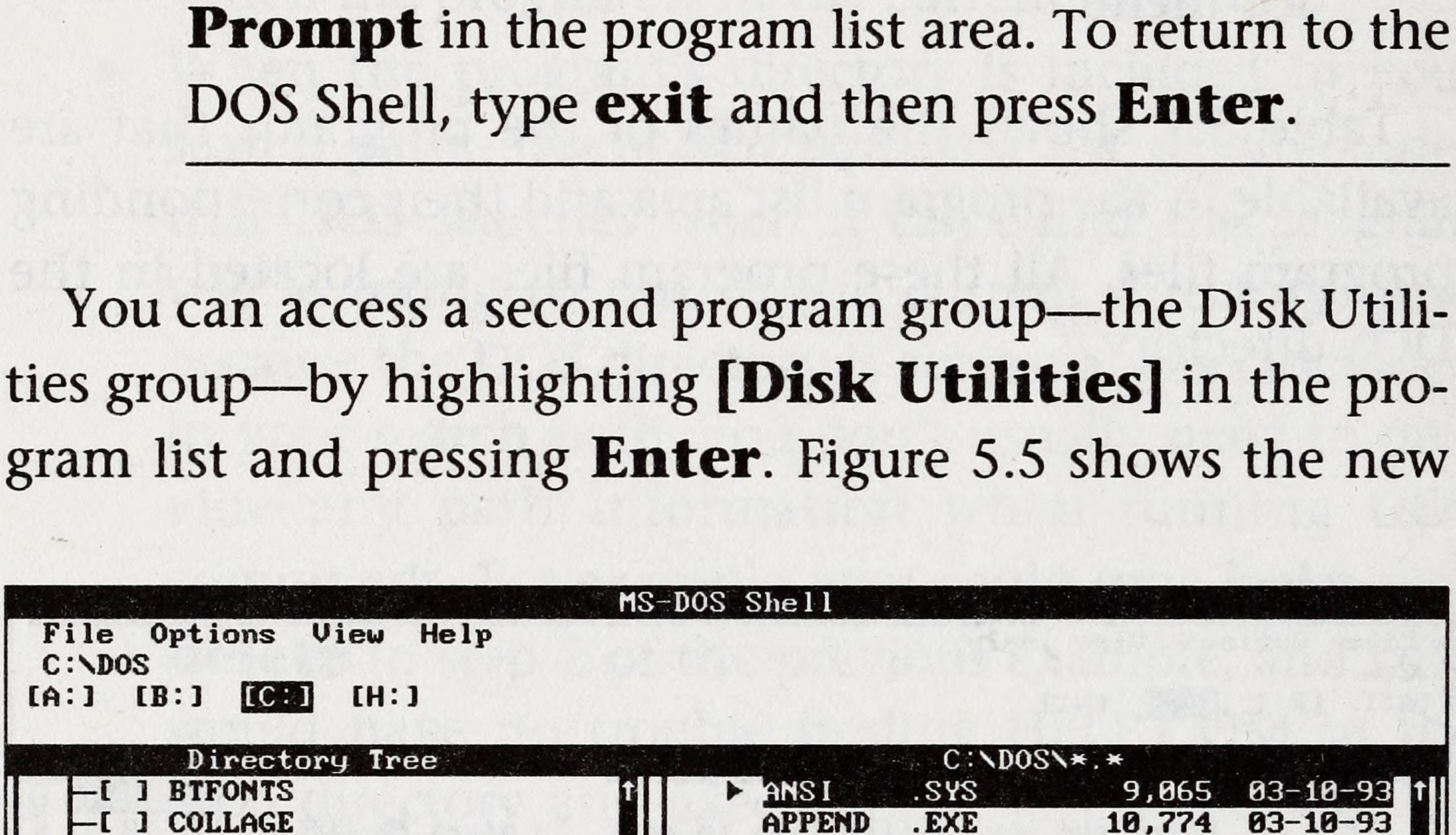


Figure 5.5 [Disk Utilities] in the program list area

Table 5.2 Disk Utilities Group Programs

Name	Program File
Disk Copy	DISKCOPY.COM
Backup Fixed Disk	BACKUP.EXE
Restore Fixed Disk	RESTORE.EXE
Quick Format	FORMAT.COM (with the /q switch)
Format	FORMAT.COM
Undelete	UNDELETE.EXE

program names that appear in the program list. Each of these names also corresponds to a DOS utility, as shown in Table 5.2. To return to the Main program group, double-click on **[Main]** at the top of the program list or highlight it and press **Enter**.

Lesson 10 will explain how to add your own programs to the Main and Disk Utilities program groups. This lesson will also teach you how to create your own program groups.

◆ *Lesson* ◆

6

Selecting Multiple Files for a Command

Methods of Selection

Occasionally, you'll want to select a group of files from the file list before applying a command to them. You might do this, for instance, before copying a group of files from one directory to another.

The DOS Shell provides three main ways to select files using the keyboard or the mouse. You can:

- select consecutive files
- select nonconsecutive files
- select all the files in the current directory

Selecting Consecutive Files

To select a group of consecutive files—that is, files that are listed next to one another in the file list—hold down the Shift key as you press ↓ or ↑ to extend the selection.

For example, suppose you're working in the DOS Shell and you want to select all the files in the DOS directory that begin with the letter F. Here's how:

1. Press **Tab** until the directory list area is active. Then press **d** until the **DOS** directory is selected. Press **Tab** to move to the file list area.
2. Press **f** to move to the first file that begins with F, **FASTHELP.EXE**.
3. Press **Shift** and hold it down as you press ↓ five times to highlight all the files that begin with F. (If you go too far, just press ↑ while you continue to hold **Shift** down.) Figure 6.1 shows how your screen should appear.

Tip

The Options Select Across Directories command—available only when the file list area is active—allows you to extend the selection beyond the current directory. Choosing this command once turns this option on; choosing this command again turns it off.

If you were to select the File Copy command now, you'd see that all the files were listed one after the other in the From: location, each separated by a space. All you would need to do to copy the files to another directory would be to provide the name of the directory for the To: location; for example, **c:**.

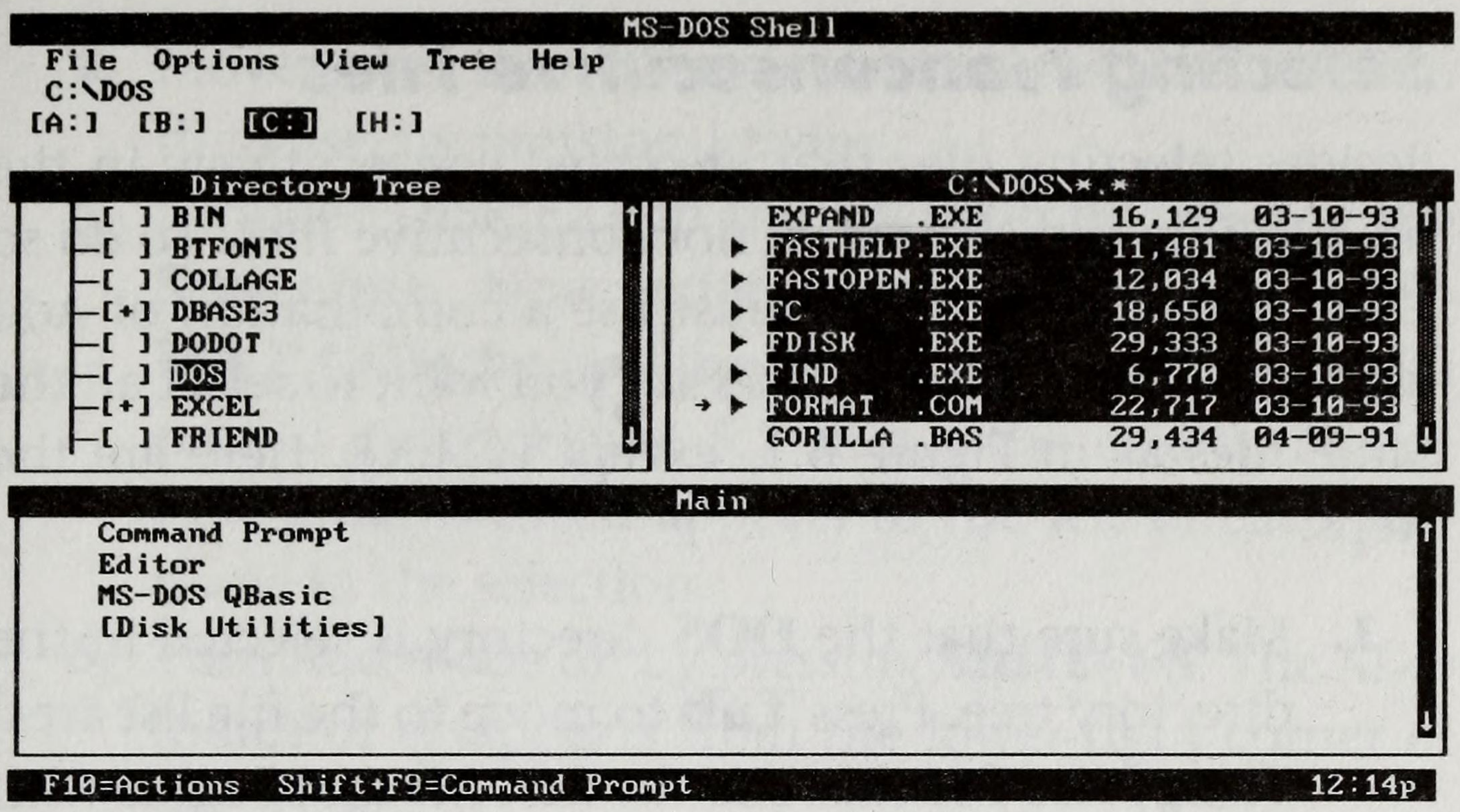


Figure 6.1 Selecting consecutive files using Shift

Tip

After highlighting a group of consecutive files with Shift, you can eliminate the selection either by pressing Ctrl+\ or by releasing the Shift key and using ↑ or ↓ to move the highlight to another file.

To select a group of consecutive files using the mouse, start by clicking on the first file you want included in the group—for example, FASTHELP.EXE in Figure 6.1. Then hold down the Shift key as you click on the last file you want in the group—FORMAT.COM in this case. All the files between (and including) the first and last files you clicked on will immediately appear highlighted.

Selecting Nonconsecutive Files

Besides selecting files that are listed consecutively in the file list, you can also select nonconsecutive files. To do so using the keyboard, you must use a combination of Add mode and the Spacebar. Let's say you want to select all the same files as in Figure 6.1, except FC.EXE. Here are the steps:

1. Make sure that the **DOS** directory is selected in the directory tree. Press **Tab** to move to the file list area. Then press **f** to move to the first file that begins with F, **FASTHELP.EXE**.
2. Press **Shift** to begin the selection. Then press ↓ to extend the selection to **FASTOPEN.EXE**. Both FASTHELP.EXE and FASTOPEN.EXE are now highlighted.
3. Extend the selection to noncontiguous files by pressing **Shift+F8** to turn **Add** mode on. You'll see ADD displayed in the lower-right corner of the DOS Shell window.

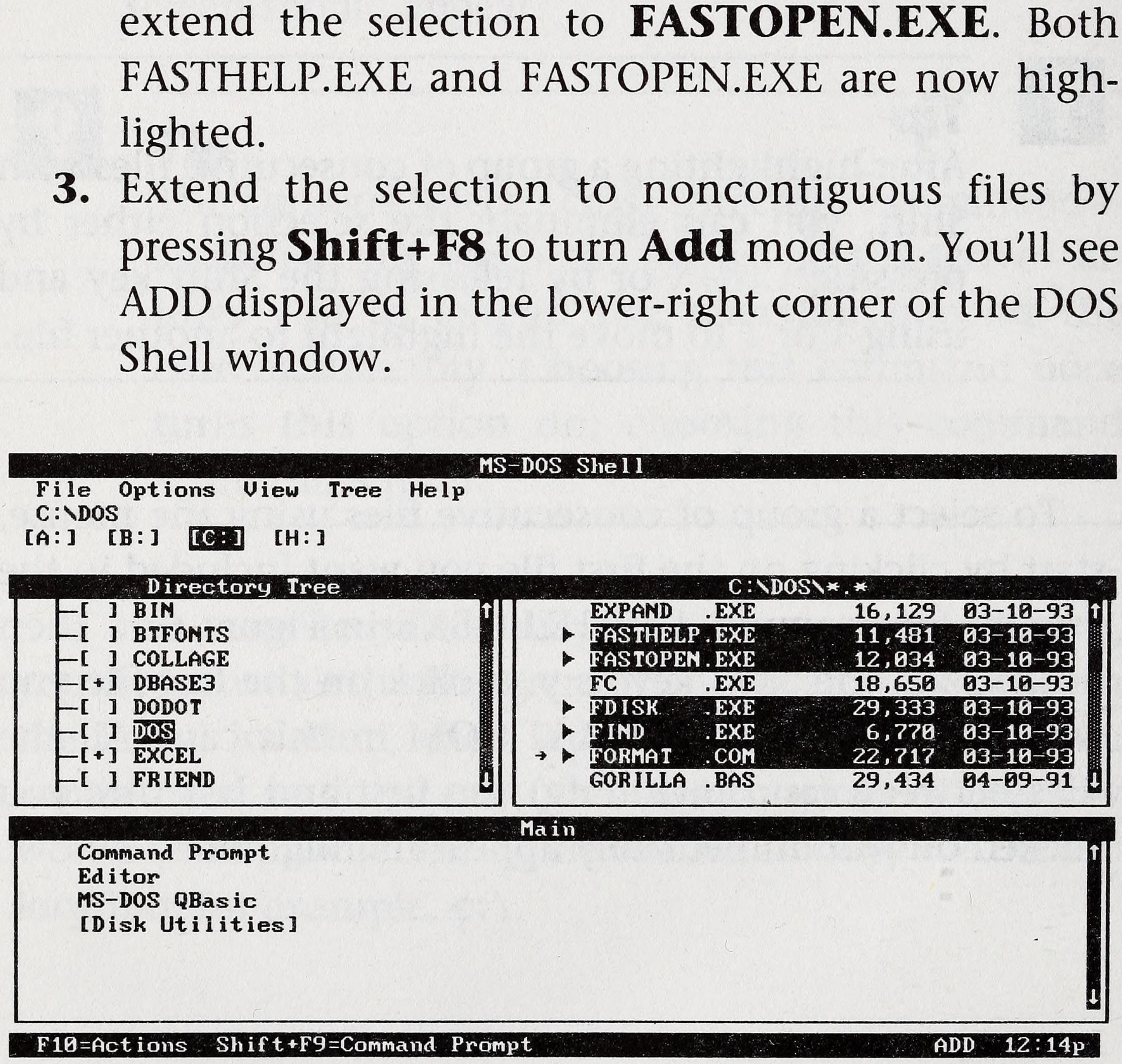


Figure 6.2 Selecting nonconsecutive files

4. Move to the next file to be selected, **FDISK.EXE** in this case, by pressing ↓ twice.
5. Include FDISK.EXE in the selection by pressing the **Spacebar**. Now extend the selection to **FORMAT.COM** by keeping **Shift** depressed while you press ↓ twice. Figure 6.2 shows the results. Notice how an arrowhead appears to the left of each filename in the selection.
6. Turn Add mode off by pressing **Shift+F8**. The ADD indicator disappears from the lower-right corner of the screen.

Tip

When Add mode is on, the Spacebar acts as a toggle key. This means you can use the Spacebar to add the highlighted file to, or remove it from, the current selection.

To select nonconsecutive files using the mouse, start by clicking on the first filename you want selected. Then keep **Ctrl** depressed as you click on each additional file you want included in the selection. To remove a file from the current selection, click on it again.

Selecting All the Files in a Directory

The quickest way to select all the files in the file list is to press **Ctrl+/**, or use the **File Select All** command. To cancel the current file selection, press **Ctrl+** again, or choose the **File Deselect All** command.

◆ *Lesson* ◆

7

Different Ways of Viewing Files and Directories

Changing the DOS Shell Screen Arrangement

When you first load the DOS Shell, the position of the directory tree, file list, and program list may seem cast in concrete to you. In reality, though, you can easily change the arrangement of these areas using the View menu. To see how this works, try this:

1. Choose the **View Single File List** command. Figure 7.1 shows the view you'll see. Notice that the screen still contains the directory tree on the left and the file list on the right. The program list, though, no longer appears.
2. Change the view again by choosing the **View Program List** command. This view displays only the program list area that you normally see at the bottom of the DOS Shell.
3. Return to the default appearance and position of the directory tree, file list, and program list by selecting the **View Program/File Lists** command. Or try

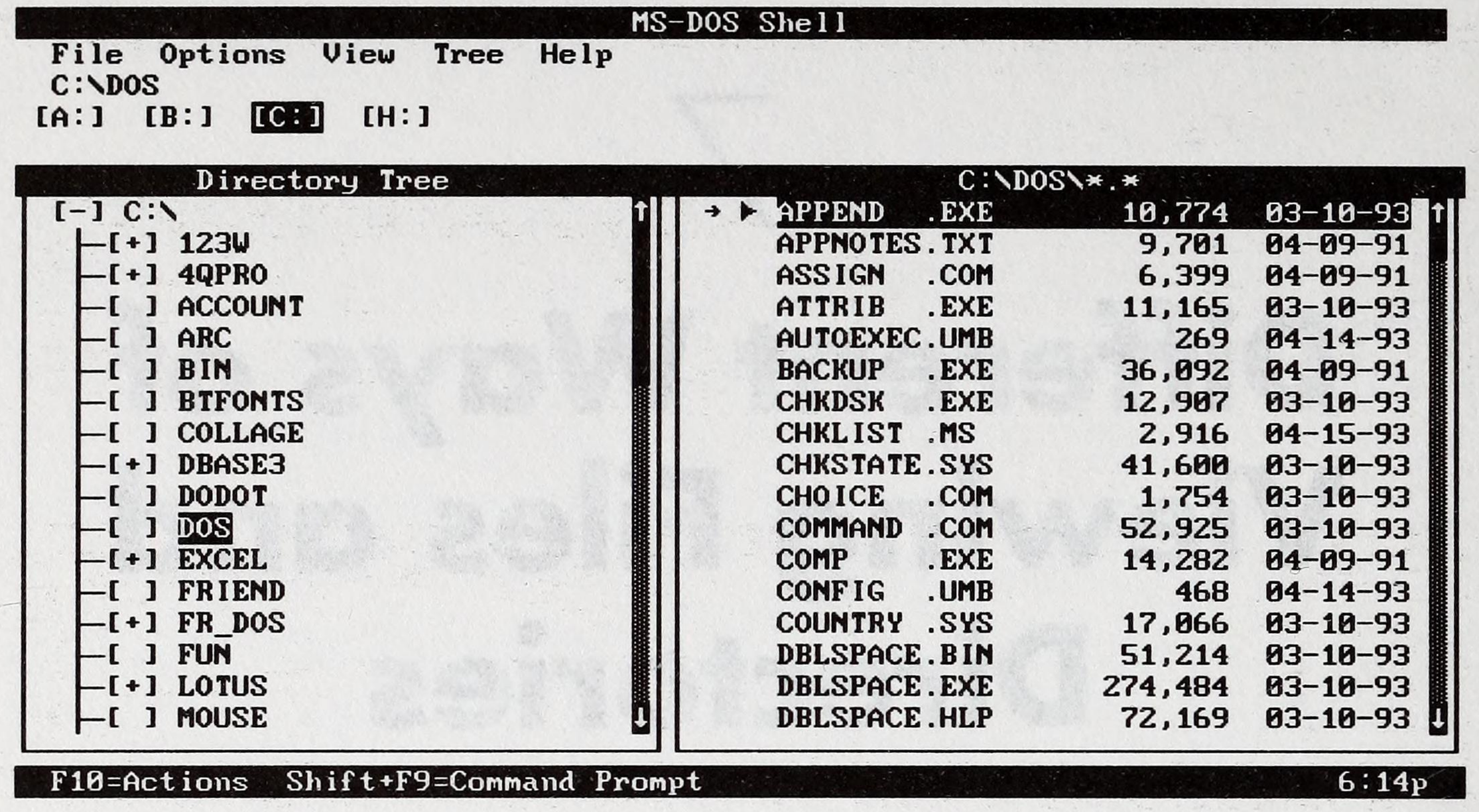


Figure 7.1 The Single File List view

one of the other views discussed below by choosing another View command.

Any new view setting is automatically saved to disk so that you'll see this new view the next time you work in the DOS Shell.

Dual File Lists View

When copying and moving files with the mouse, it's often handy to see two directory trees and file lists at the same time. That way, you can more easily drag-and-drop files between them. To see this type of view, you use the **View Dual File Lists** command. Figure 7.2 shows how the DOS Shell looks with this configuration.

After using the View Dual File Lists command for the first time during a session, you'll want to change the directory selected in the second (lower) directory tree. In Figure 7.2, for example, the top view displays the DOS directory; the bottom view shows the C:\ root directory.

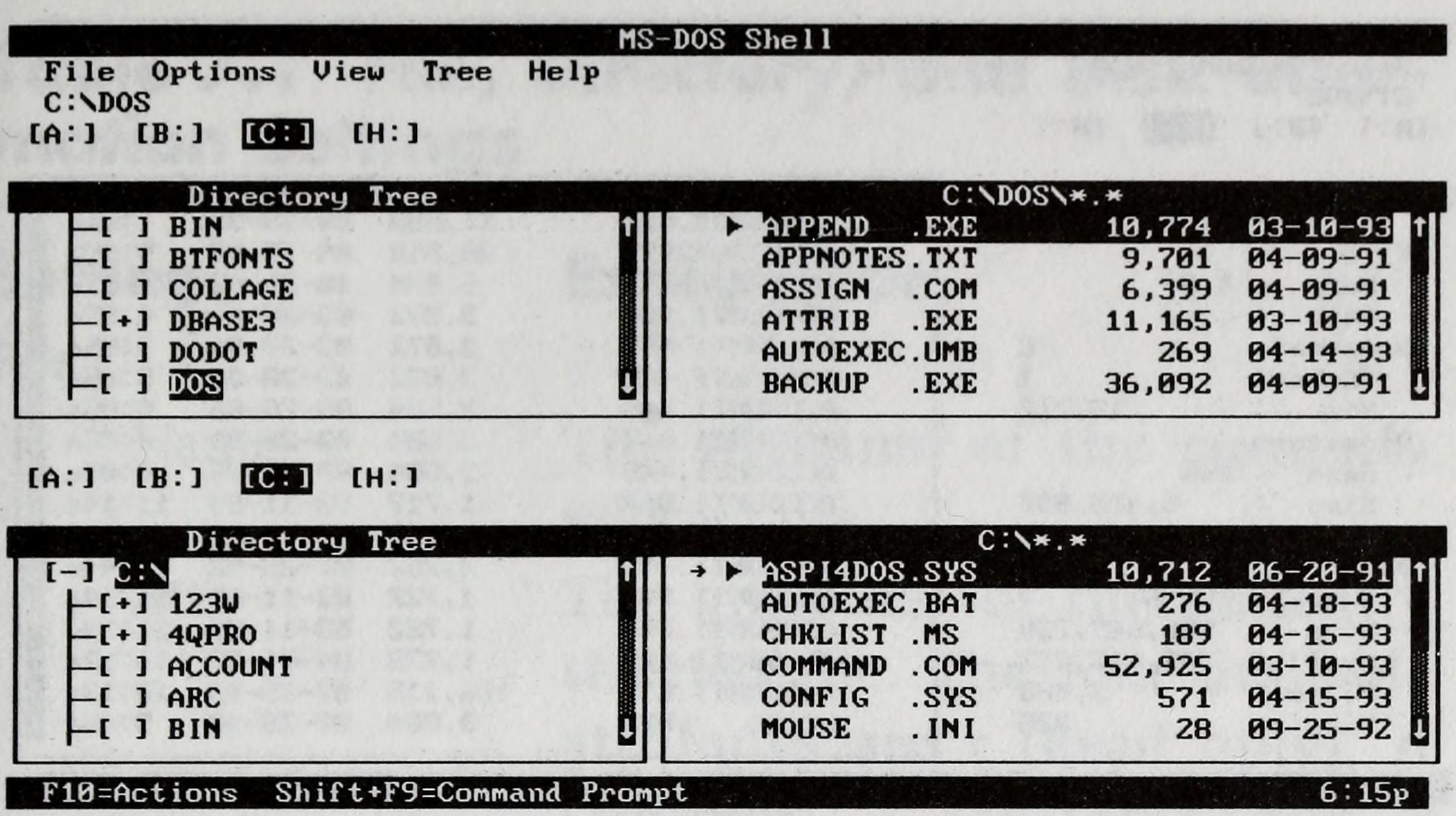


Figure 7.2 The Dual File Lists view

All Files View

By using the **View All Files** command, you can have the DOS Shell show one all-encompassing list of files from the current drive. The files are sorted alphabetically regardless of their directories. Figure 7.3 shows an example of this view. Notice that the first two files in the list have the same filename. They are located in different directories, however. The directory for the currently selected file, C:\DOS, is shown below the menu bar at the top of the screen. This line is updated with the appropriate directory information whenever you highlight a new file.

Tip

The All Files view is the only way to see the time that a file was created from within the DOS Shell. The time appears in the farthest right column of the screen.

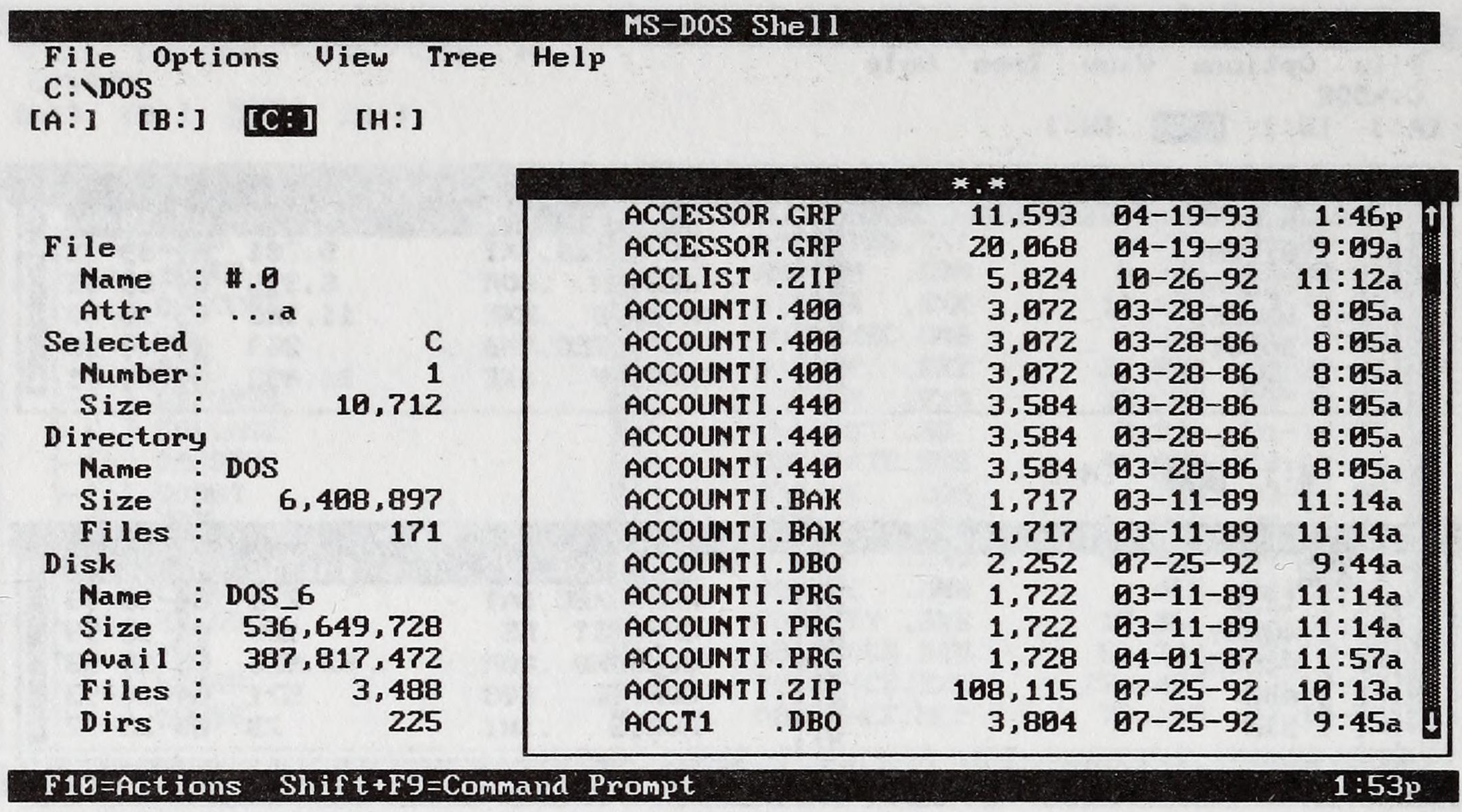


Figure 7.3 The All Files view

The left side of the screen also shows an assortment of file, directory, and disk drive information. Table 7.1 gives an explanation of the settings.

Note: The file and disk size information in the All Files view is reported in bytes, where a byte is approximately one character. You're bound to see files and disk sizes reported elsewhere in kilobytes or megabytes, so here's the conversion:

1 kilobyte (K) = 1,024 bytes
1 megabyte (MB) = 1,024 kilobytes or
1,048,576 bytes

Tip

Another way to see file and disk information is to select the **Options Show Information** command. (You can use this command with all views except the All Files view.)

Table 7.1 File, Directory, and Disk Information Settings

Setting	Explanation
File	
Name	The filename of the currently selected file.
Attr	The attributes of the currently selected file. The four available attributes are r (Read only), h (Hidden), s (System), and a (Archive). If an attribute is not set, it appears as a . (period). (See Lesson 12 for more information in file attributes.)
Selected	
Number	The number of files that are currently selected in the file list.
Size	The cumulative size (in bytes) of the currently selected files. You might check this setting, for example, to determine whether the selected files will all fit on a floppy disk at once.
Directory	
Name	The subdirectory in which the currently selected file resides.
Size	The size (in bytes) of all the files in the named subdirectory.
Files	The number of files in the named subdirectory.

(continued)

Table 7.1 *(continued)*

Setting	Explanation
Disk	
Name	The volume label of the current disk. (You set the volume label when formatting the disk. See Lesson 14.)
Size	The size of the current disk drive in bytes.
Avail	The total amount of space available on the current disk drive in bytes.
Files	The total number of files on the current disk drive.
Dirs	The total number of directories on the current disk drive.

◆ *Lesson* ◆

8

Controlling the File List

Changing File Display Options

Normally, the DOS Shell file list shows all the files in the current directory sorted by name in ascending order. By changing the file display options, though, you can:

- limit the types of files that appear by changing the file specifier
- include hidden and system files
- change the sort order of files

Limiting the Types of Files Displayed

Suppose you want the DOS Shell to display only .EXE files in its file list. To limit the file list to this type of file, you must change the file specifier. The *file specifier* is a pattern of characters that the DOS Shell uses to filter filenames. Any filenames that do not match the file specifier are omitted from the file list.

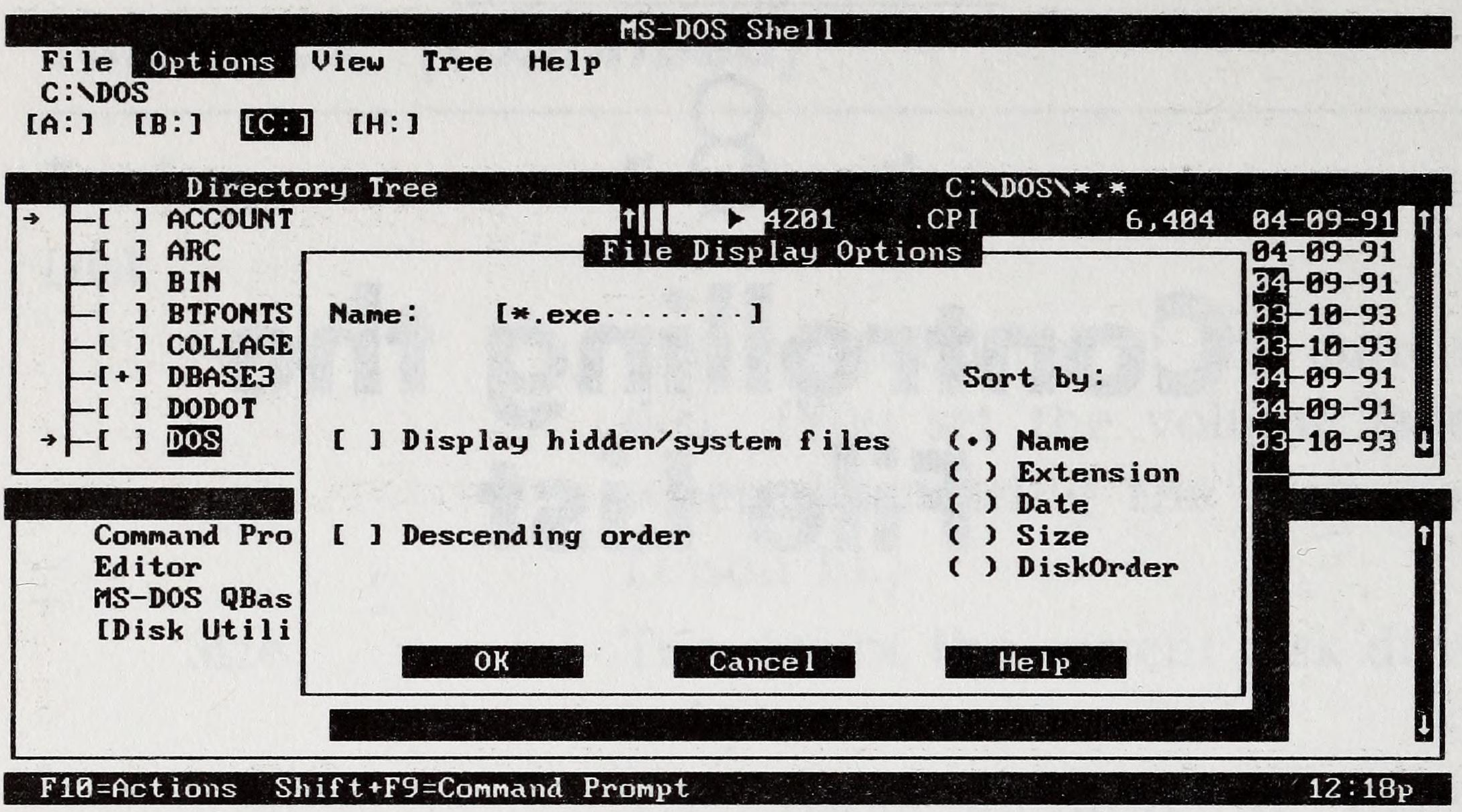


Figure 8.1 Options File Display Options box

To display only .EXE files, you must establish a file specifier of *.exe by performing the following steps:

1. Press **Tab** until the directory list area is active. Then press **d** until the **DOS** directory is selected.
2. Select the **Options File Display Options** command. Figure 8.1 shows the dialog box you see. The default setting *.* causes all file types to be displayed.
3. Type ***.exe** in the Name field and press **Enter**. The DOS Shell immediately updates the file list as in Figure 8.2 to include only .EXE files from the current DOS directory. If no files are found with the extension specified—no .EXE files, for instance—you'd see the message "No files match specifier" instead.
4. To return to displaying all files in the current directory, select the **Options File Display Options** command again, type ***.*** and then press **Enter**.

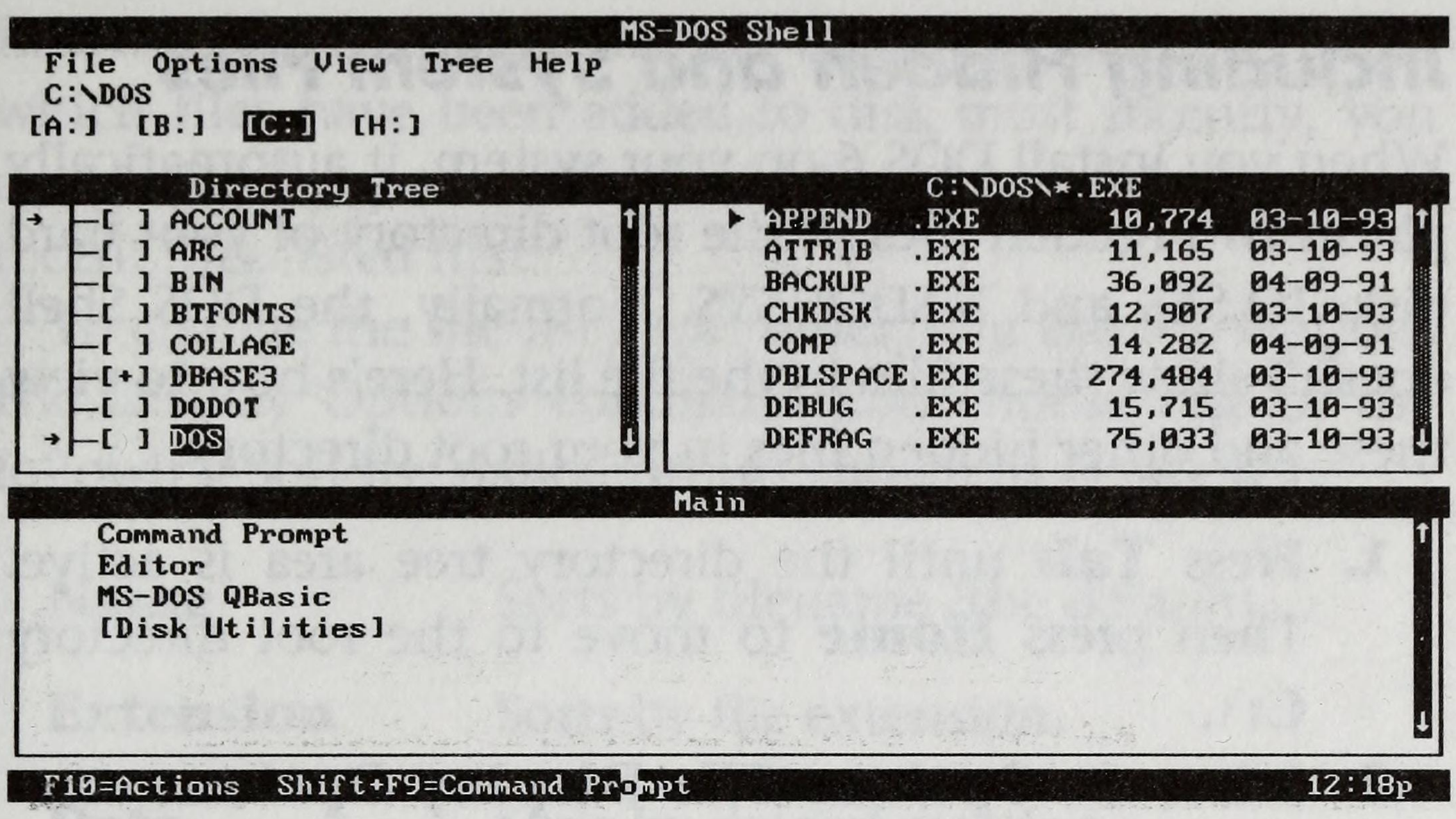


Figure 8.2 Displaying only .EXE files

THE ? AND * WILDCARD SYMBOLS

When setting the file specifier, you can use the ? and * wildcard symbols (*wildcards* for short). The ? wildcard matches any single character, and, as the previous example showed, the * wildcard matches any group of characters. Here are some more examples:

File Specifier	*Matches*
b*.*	All files that begin with the letter B—for example, B1.EXE, BLUE- SKY.WK1, and B_LEVEL3.BAK.
taxes.w*	All files named TAXES that have an extension that begins with the letter W—for example. TAXES.WB1, TAXES.W, and TAXES.WK3.
??gs.?if	Any four-letter filename that ends in GS and has a three-letter extension that ends in IF—for example, FIGS.TIF, COGS.RIF, and RAGS.BIF.

Including Hidden and System Files

When you install DOS 6 on your system, it automatically places two hidden files in the root directory of your hard disk: IO.SYS and MSDOS.SYS. Normally, the DOS Shell doesn't show these files in the file list. Here's how to view these and other hidden files in your root directory:

1. Press **Tab** until the directory tree area is active. Then press **Home** to move to the root directory **C:**.
2. Select the **Options File Display Options** command. Then press **Tab** to move to the Display hidden/system files check box.
3. Press the **Spacebar** to turn the checkbox on. Then press **Enter**. (Using the mouse, click on this checkbox.) You'll now see IO.SYS and MSDOS.SYS in the file list as well as any other hidden files.
4. To omit the hidden files from the file list, select the **Options File Display Options** command again. Press **Tab** to move to the Display hidden/system files check box. Press the **Spacebar** to turn the checkbox off and then press **Enter**.

Note: Viewing hidden files doesn't change their hidden status. For that, you'll need to change their file attributes. See Lesson 12.

Changing the Sort Order

As mentioned, the DOS Shell file list normally shows your files sorted by name in ascending order. Although this method of display is fine in most cases, occasionally you'll

find it useful to change the sort order. For example, to see which files have been added to disk most recently, you could display your files sorted by date, with the most recent files listed first.

To change the file list's sort order, you use the Options File Display Options command and choose one of the following Sort by: radio buttons shown in Figure 8.1:

Name	Sorts by filename (the default).
Extension	Sorts by file extension.
Date	Sorts by date of creation.
Size	Sorts by file size.
DiskOrder	This is the actual order in which files are placed on disk; in other words, selecting this option means that no sorting takes place at all.

In addition, you can specify whether the file list uses a descending sort order by turning on the Descending Order check box.

Suppose you want the DOS Shell to refrain from sorting your files so that you can see where on the disk they actually reside. To do so, turn on the DiskOrder radio button, as follows:

1. Select the **Options File Display Options** command. Make sure that *.* appears in the Name field.
2. Press **Tab** three times to move to the Sort by: radio buttons.
3. Press ↓ four times to turn on the **DiskOrder** radio button. (If you have a mouse, click on the DiskOrder radio button to turn it on.) Then press **Enter**.

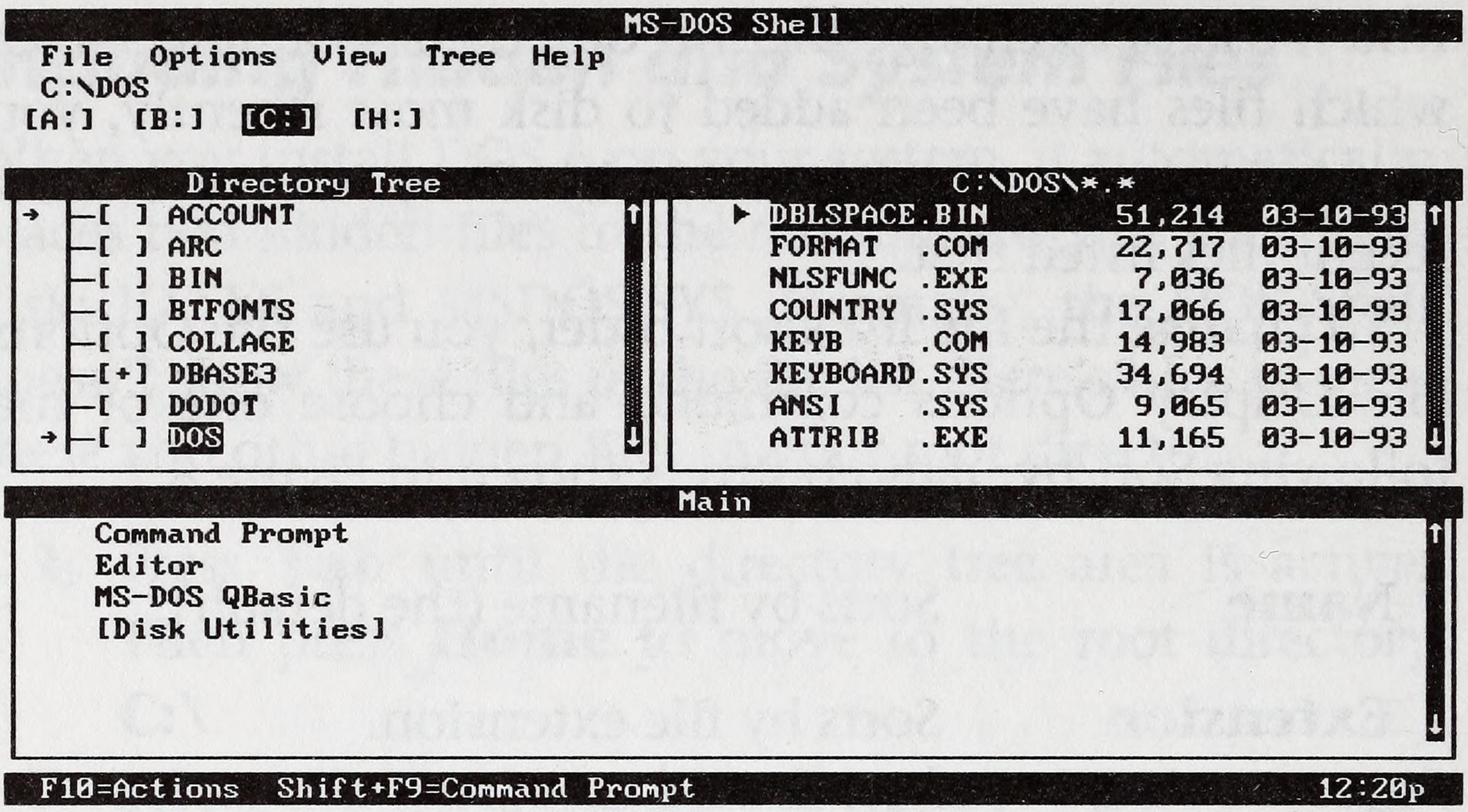


Figure 8.3 The DiskOrder radio button turned on

Figure 8.3 shows the effect of this new setting when you select the DOS directory. Notice that the files no longer appear in alphabetical order but in the order they were actually placed on the disk.

Rereading All Files and Directories

Occasionally, the files and directories listed in the DOS Shell will not reflect the true state of the disk. For example, suppose you temporarily leave the DOS Shell and execute a command at the DOS command line to create a new directory (explained in Lesson 18). When you return to the DOS Shell, the directory tree will not show the new directory you've created. To update the DOS Shell so that it shows the new directory, select the **View Refresh** command or its shortcut, **F5**. The DOS Shell will pause for a moment as it rereads all the files and directories on the current disk and then show the new directory in its updated directory tree and file list.

◆ *Lesson* ◆

9

Searching for Files

Searching for a Single File

As the number of files on your hard disk increases, you may forget where a file is stored. Rather than spending a lot of time poking around, consider using the DOS Shell File Search command instead.

Recall that in Lesson 4, you were asked to move the OS2.TXT file from the your C:\DOS directory to your root directory (C:\). You were then asked to move it back. Let's suppose that something went wrong during that lesson and now you can't find the OS2.TXT file anywhere. From the DOS Shell, here's how to search for it:

1. Select the **File Search** command. DOS Shell displays the dialog box in Figure 9.1.
2. Type **os2.txt** in the **Search for** text box.
3. Make sure the **Search entire disk** check box is turned on. That way the DOS Shell searches your entire drive C. (If you turn off this check box, only the current directory will be searched. The current directory is indicated at the top of the Search File dialog box.)

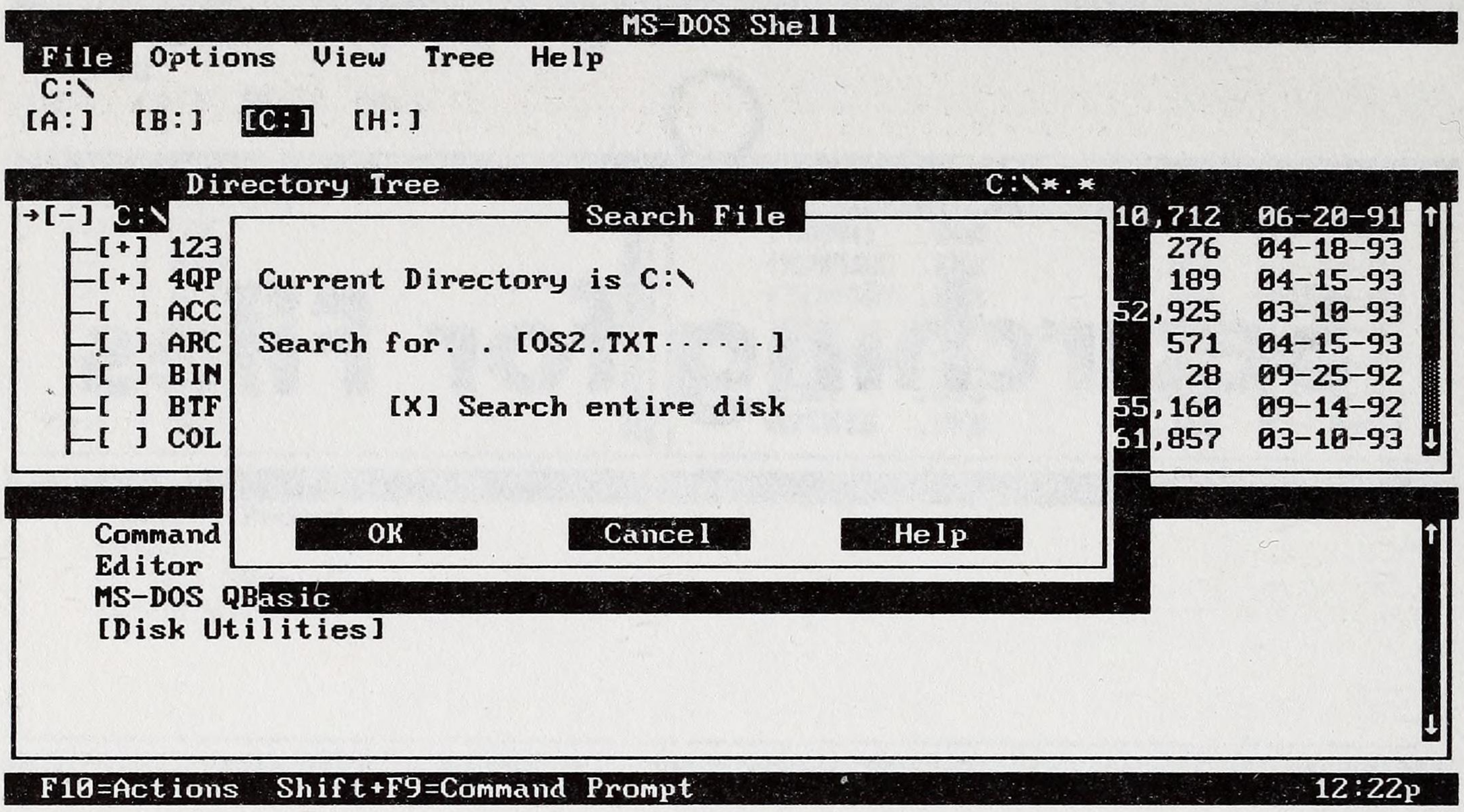


Figure 9.1 Searching for files

4. Press **Enter** to begin the search. DOS Shell hesitates for a moment and then displays the screen in Figure 9.2 showing the location of the OS2.TXT file.

⇨ ***Note:*** If you're running DOS 5, you'll see the message "No files match file specifier" instead because no OS2.TXT exists in DOS 5.

5. Note the location of OS2.TXT, then press **Esc** to return to the DOS Shell. (Alternatively, you could also perform an operation on a file in the search list, such as copy, move, rename or delete. Or, if it's a program (.COM or .EXE) file, you could select the file to run the program.)

Searching for a Group of Files

You can also search for groups of files with related filenames or extensions by using wildcards. For example,

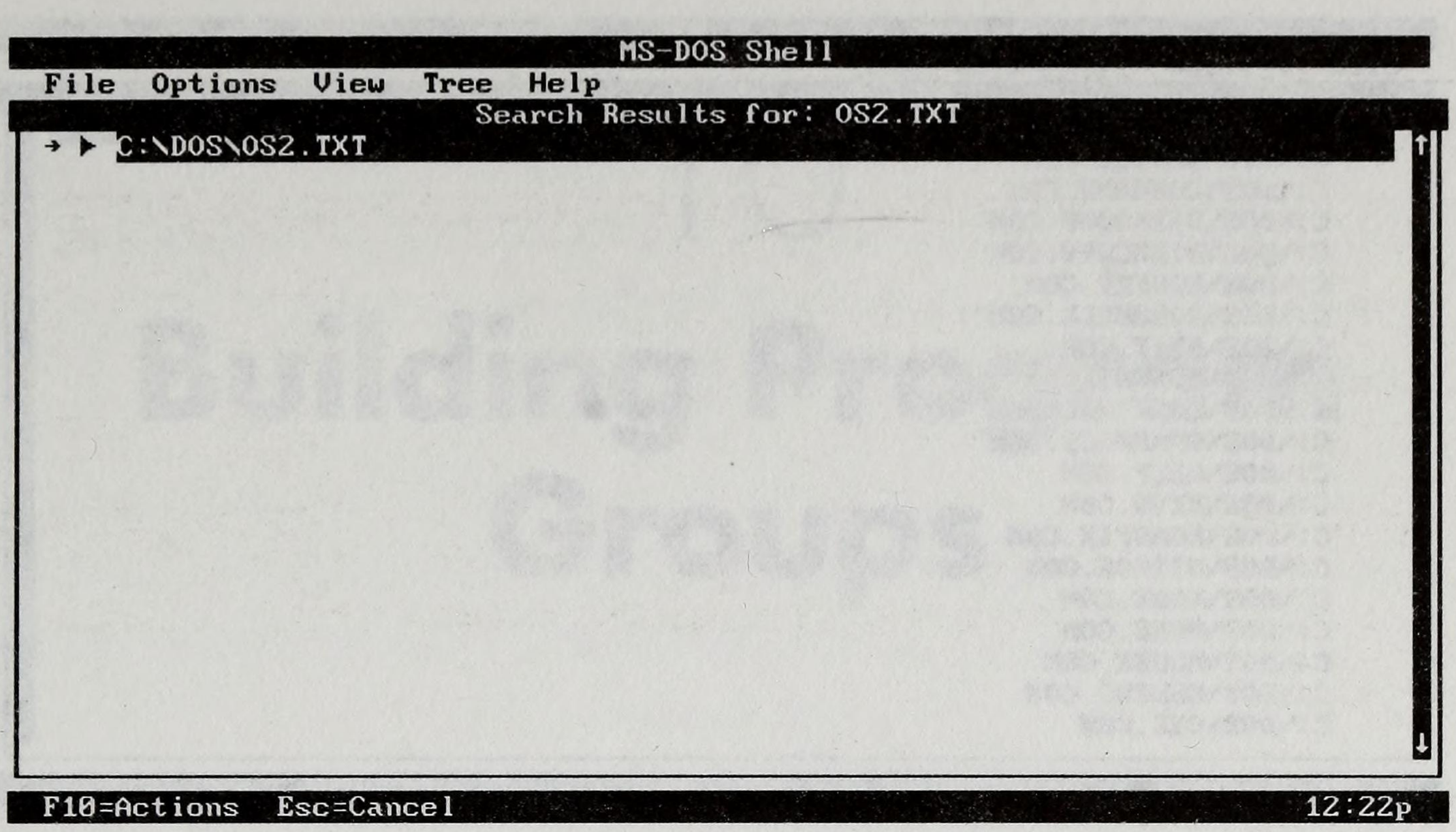


Figure 9.2 OS2.TXT in the C:\DOS directory

suppose you want to locate all the files in your DOS directory that have a .COM extension. To do this, perform the following steps:

1. Press **Tab** until the directory list area is active. Then press **d** until the **DOS** directory is selected.
2. Select the **File Search** command to access the Search File dialog box shown in Figure 9.1.
3. Type ***.com** in the Search for text box. This wild card pattern will select all files with a .COM extension.
4. Move to the Search entire disk check box by pressing **Tab**. Then press the **Spacebar** to turn this option off.
5. Press **Enter** to begin the search. The DOS Shell hesitates for a moment and then displays a screen like the one in Figure 9.3 showing all the .COM files residing in the DOS directory.

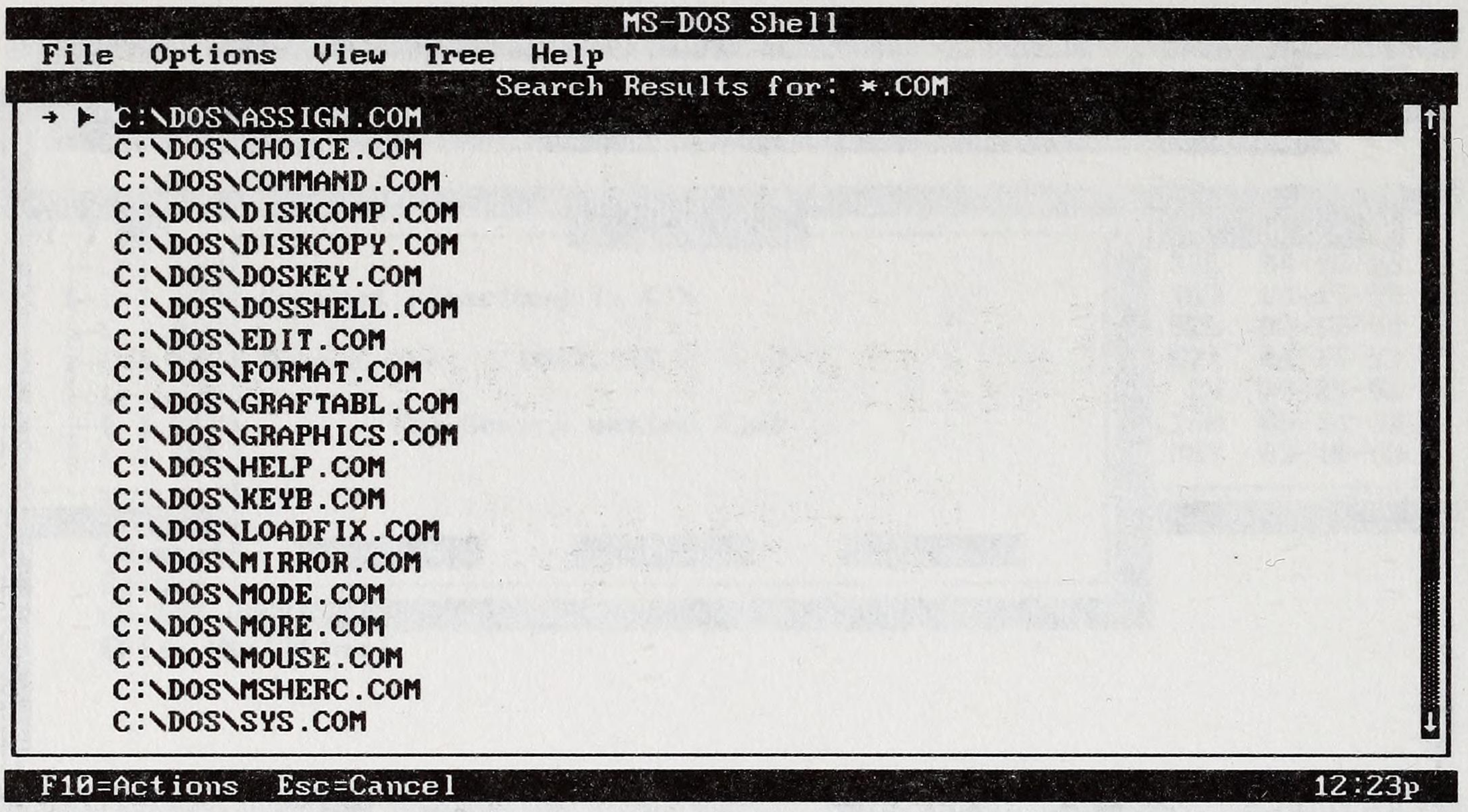


Figure 9.3 Searching for a group of files

◆ *Lesson* ◆

10

Building Program Groups

Understanding Program Groups

As you learned in Lesson 5, a program group is a collection of program names that appear in the program list area. The DOS Shell has two program groups: the Main group, which is displayed automatically when you start up the DOS Shell, and the Disk Utilities group, which appears when you select [Disk Utilities] in the program list.

Although these two program groups appear to be permanent fixtures in the DOS Shell, actually you can manipulate them to your own liking. In fact, you can perform any of the following operations related to program groups:

- Create a new program group.
- Add program items to a group.
- Rearrange program groups.
- Change the properties of a program item.

Contents of Existing Program Groups

Before creating new program groups and adding program items to them, let's review the contents of the existing program groups and how to navigate among them. The items in the [Main] program group are:

Command Prompt	Runs the MS-DOS command interpreter COMMAND.COM, taking you to the DOS command line. (See Lesson 17.)
Editor	Starts the MS-DOS Editor EDIT.COM, which lets you view and edit ASCII files. (See Lessons 13 and 28.)
MS-DOS QBasic	Starts the MS-DOS QBasic utility QBASIC.EXE.

The [Disk Utilities] group contains the following program items:

Disk Copy	Runs the Diskcopy utility DISKCOPY.COM, which copies floppy disks. (See Lesson 15.)
Backup Fixed Disk	Runs the Backup utility BACKUP.EXE, which backs up files. (See Lesson 25.)
Restore Fixed Disk	Runs the Restore utility RESTORE.EXE, which restores backed up files. (See Lesson 26.)
Quick Format	Runs the Format utility FORMAT.COM with the /q switch,

	which erases the contents of a disk. (See Lesson 14.)
Format	Runs the Format utility FORMAT.COM, which formats a disk. (See Lesson 14.)
Undelete	Runs the Undelete utility UNDELETE.EXE, which recovers deleted files. (See Lesson 23.)

Working in the Program List Area

Before you begin working with program groups, you'll want to make sure that the program list area is active. (If another area of the DOS Shell is active, keep pressing Tab.) That's because the commands that apply to program groups differ from the other areas of the DOS Shell. Figure 10.1, for instance, shows the File commands you see when the program list area is active.

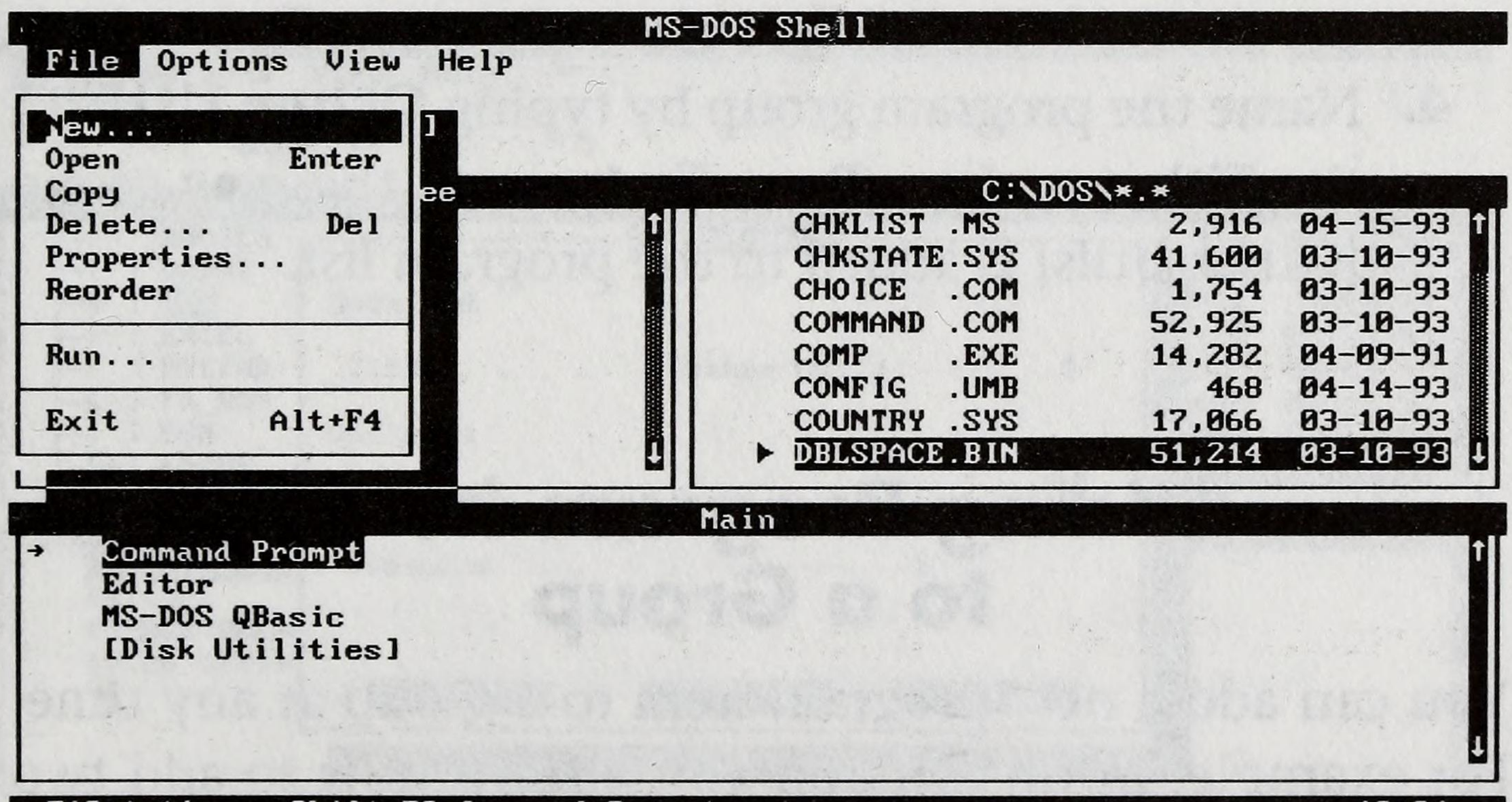


Figure 10.1 The program list area activated in the File menu

Remember from Lesson 5 that you can run a program item such as the Editor by highlighting it and then pressing Enter. If you're using the mouse, simply double-click on the item you wish to run.

You use one of the same techniques to open a program group like the [Disk Utilities]. To close a program group, use ↑ to highlight [Main] at the top of its program list and then press Enter. If you're using a mouse, just double-click on [Main].

Creating a New Program Group

To see how to create a new program group in the DOS Shell, try creating one called "Other Utils":

1. Press **Tab** until the program list area is active.
2. Select the **File New** command. You'll see the New Program Object dialog box in Figure 10.2.
3. Turn on Program Group and turn off Program Item by pressing the **Spacebar**. Press **Enter** and you'll see the Add Group dialog box shown in Figure 10.3.
4. Name the program group by typing **Other Utils** in the Title text box. Press **Enter** and the new group [Other Utils] is added to the program list.

Adding Program Items to a Group

You can add a new program item to a group at any time. For example, in this section you'll learn how to add two program items to the Other Utils group created in the previous section. The first program item will be called

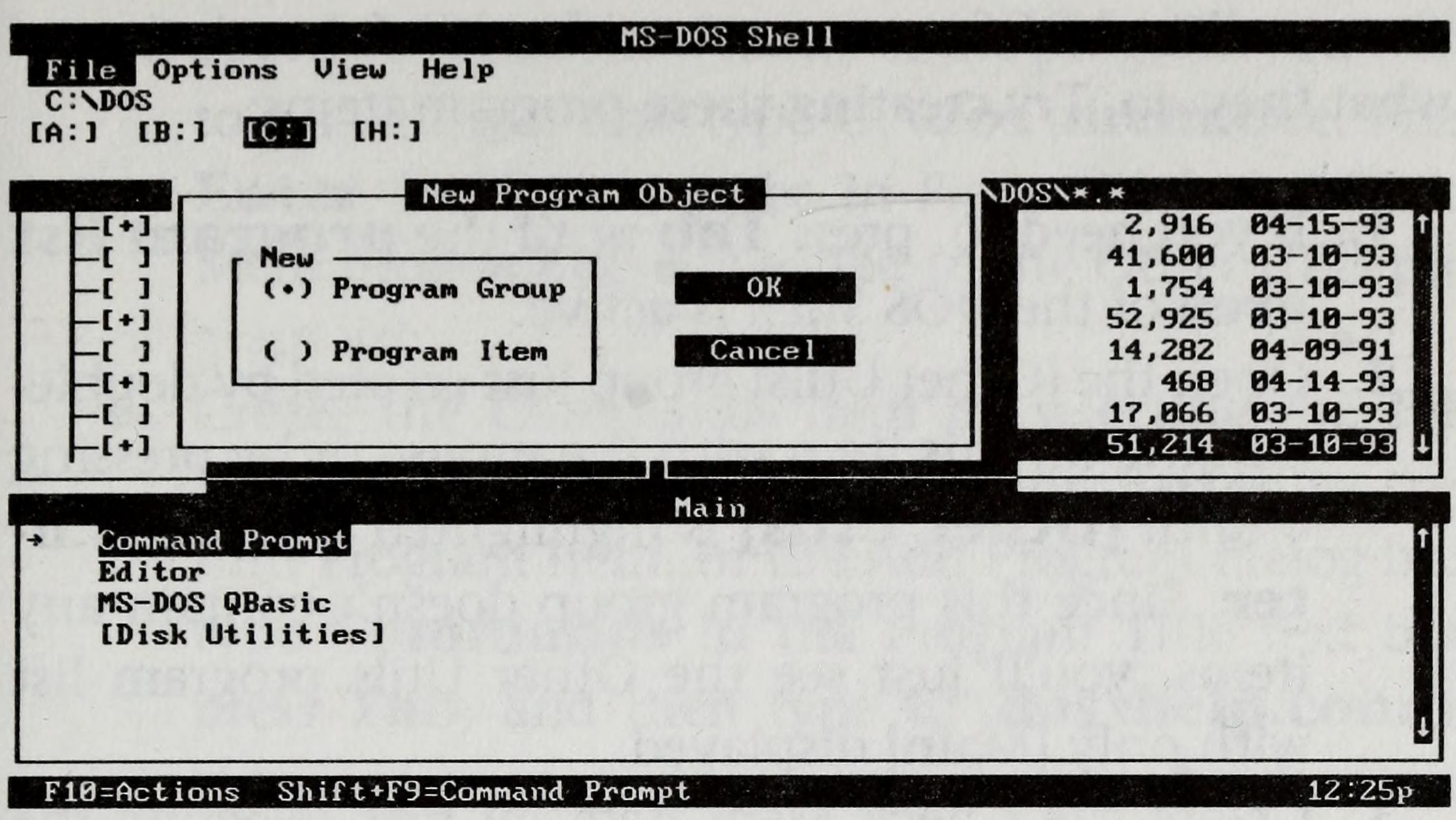


Figure 10.2 The New Program Object dialog box

Check Mem; it will run the Mem utility (MEM.EXE) located in your C:\DOS directory, which shows you how much memory you have available. The second program item will be called Commands; it will run the Help utility (HELP.EXE) located in the C:\DOS directory, which

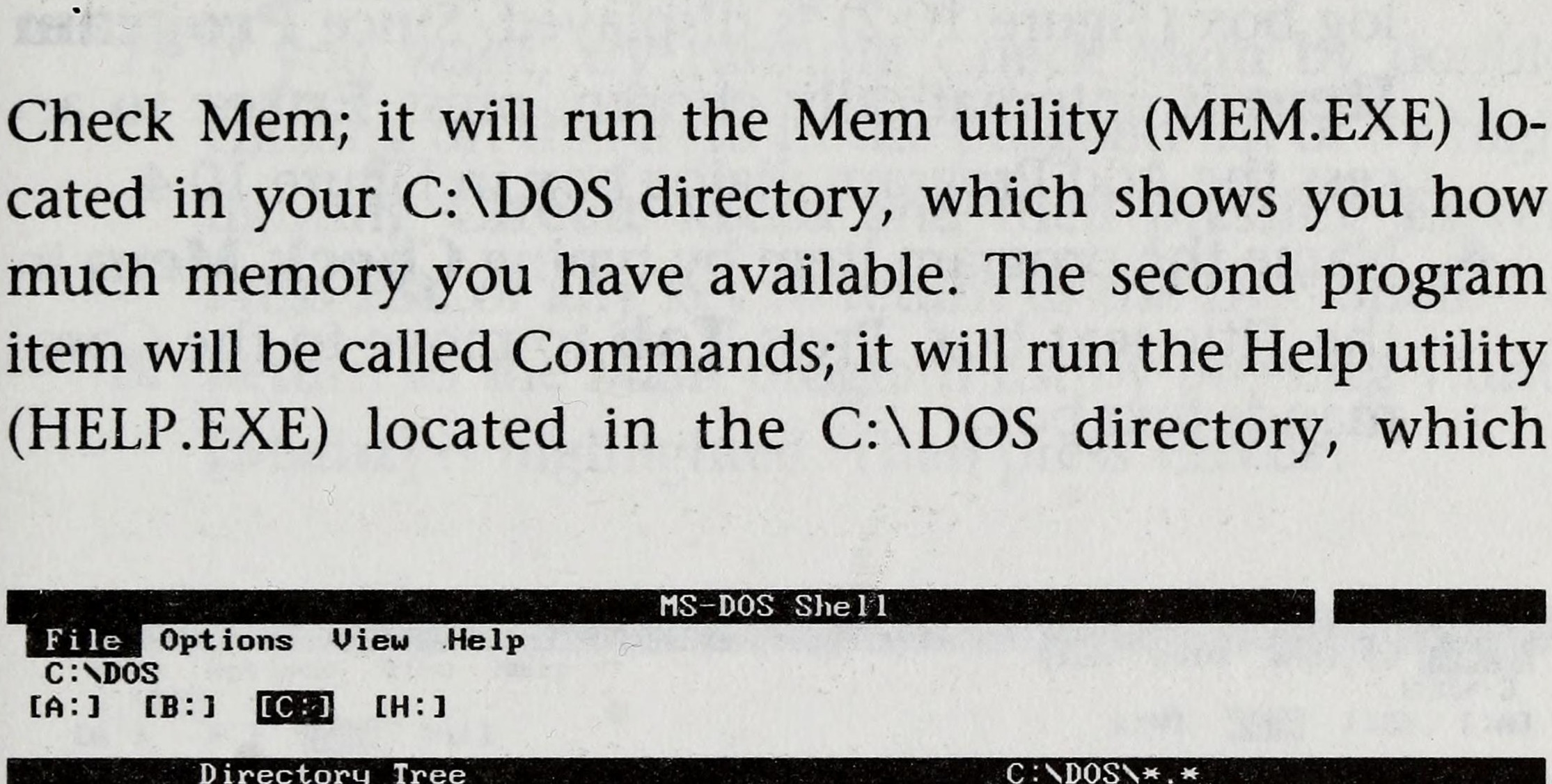


Figure 10.3 The Add Group dialog box

shows a list of DOS commands with a brief description of what they do. Try creating these program items:

1. If you need to, press **Tab** until the **program list area** of the DOS Shell is active.
2. Open the [Other Utils] group just created by double-clicking on this item with the mouse or by pressing ↓ until **[Other Utils]** is highlighted and then **Enter**. Since this program group doesn't contain any items, you'll just see the Other Utils program list with only [Main] displayed.
3. Create the Check Mem item by first selecting the **File New** command. The New Program Object dialog box (Figure 10.2) is displayed. Since **Program Item** is automatically chosen, press **Enter** to access the Add Program dialog box in Figure 10.4.
4. Name the program item by typing **Check Mem** in the Title text box. Press **Tab** to move to the Commands text box.

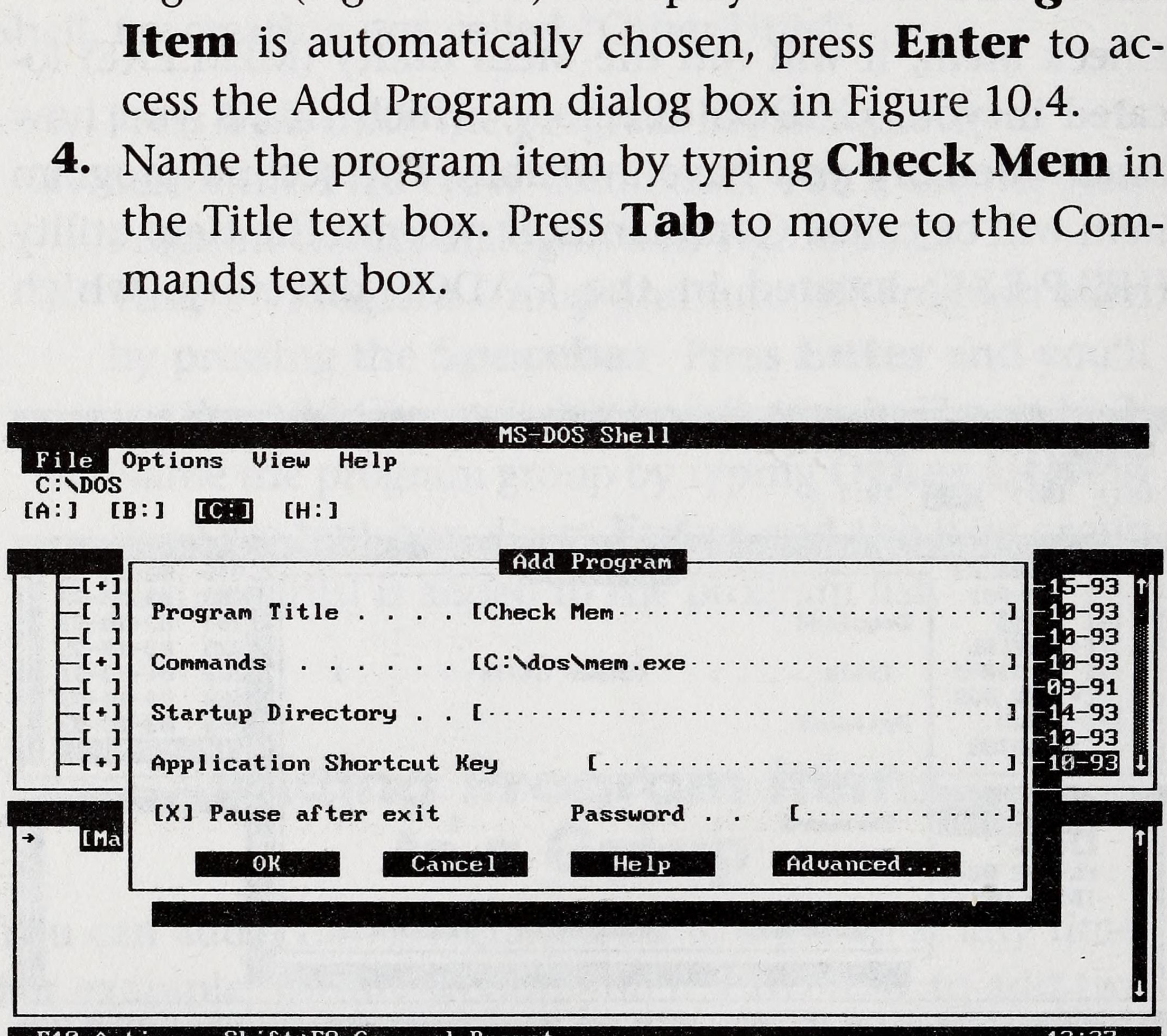


Figure 10.4 The Add Program dialog box

5. Specify the path and name of the program you want to run. In this case type **C:\dos\mem.exe**. Press **Enter** and you can see in Figure 10.5 the Check Mem program item is added to the Other Utils program list.
6. Create the Commands item by selecting the **File New** command and then pressing **Enter** to confirm Program Item. In the Add Program dialog box, type **Commands** in the Program Title text box, press **Tab**, and then type **C:\dos\help.com** in the Commands text box. Press **Enter** and the Commands program item shown in Figure 10.5 is added to the Other Utils program list.
7. If you want, try running Check Mem by double-clicking on this item in the program list or by highlighting **Check Mem** and then pressing **Enter**. Press **Esc** or any key to return to the DOS Shell.
8. Return to the Main program list by pressing ↑ until **[Main]** is highlighted. Then press **Enter**.

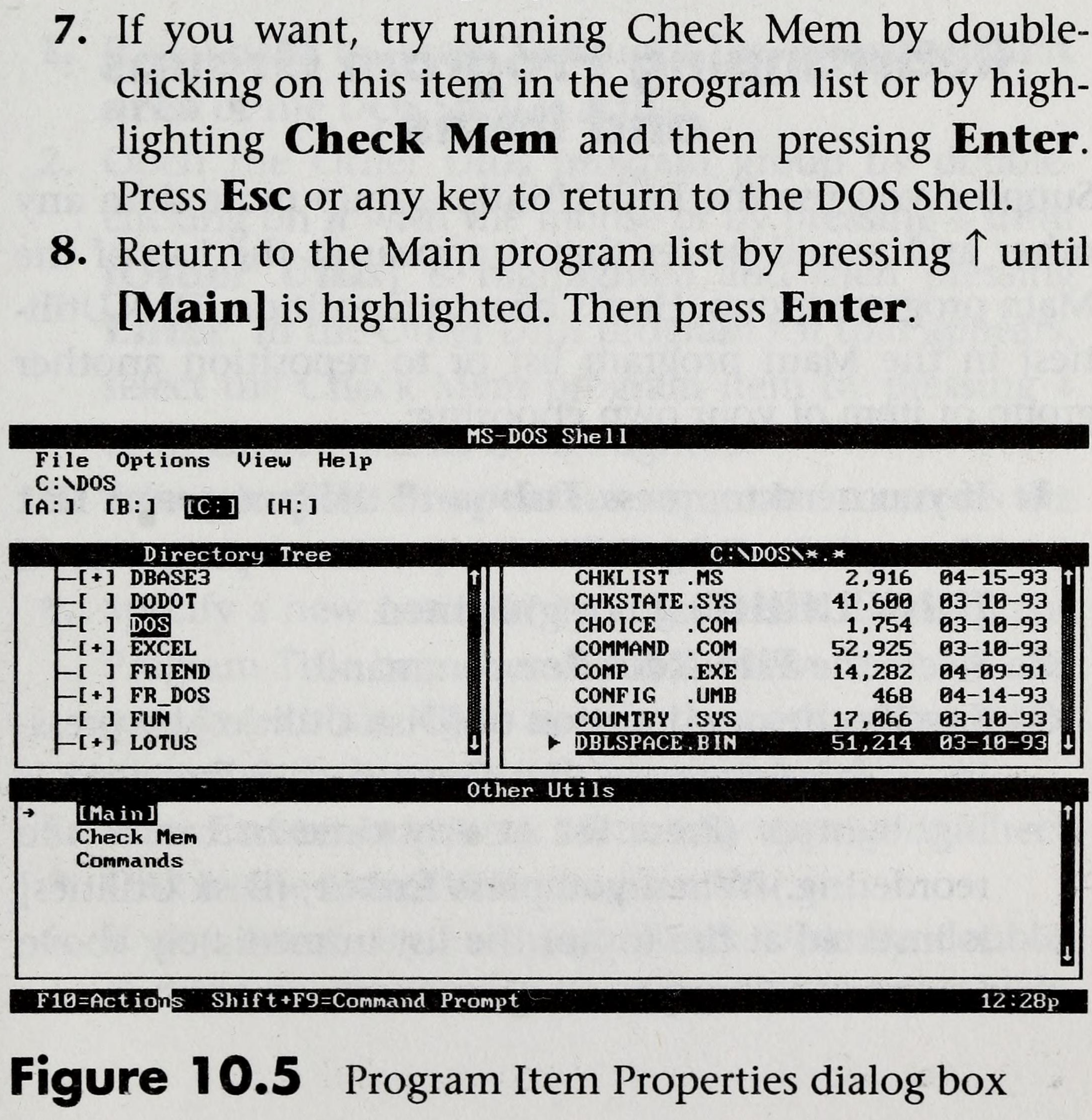


Figure 10.5 Program Item Properties dialog box

Tip

DOS 6 includes an enhanced mem command that you can use at the DOS command prompt. See Lesson 21.

Deleting Program Groups and Items

To delete a program item, select it and press Del. Del, however, only deletes an *empty* program group, so you must first delete all of the program items in the group.

Rearranging Program Groups and Items

Suppose you use the Disk Utilities group more than any other and would prefer that it appear at the top of the Main program group. Here's how to reposition [Disk Utilities] in the Main program list or to reposition another group or item of your own choosing:

1. If you need to, press **Tab** until the **program list area** of the DOS Shell is active. Then press ↓ until **[Disk Utilities]** is highlighted.
2. Select the **File Reorder** command.
3. Specify the new location of [Disk Utilities] by pressing ↑ three times so that **Command Prompt** is highlighted. (Press Esc at any time to cancel the reordering.) When you press **Enter**, [Disk Utilities] is inserted at the top of the list immediately above Command Prompt.

Changing the Properties of a Program Item

The properties of a program item also can be changed—for example, its name, the program it runs, and so on. To see how, let's change the program item Check Mem you created in the previous section, which is part of the Other Utils program group also created earlier. Currently, Check Mem runs the Mem utility (MEM.EXE) which show how much memory you have available. Let's now rename and modify Check Mem to run an entirely different utility instead, Chkdsk (CHKDSK.EXE), to display information about your hard and floppy disks:

1. If you need to, press **Tab** until the **program list area** of the DOS Shell is active.
2. Open the Other Utils program group by double-clicking on it with the mouse or by pressing ↓ until **[Other Utils]** is highlighted and then pressing **Enter**. In the Other Utils program list that appears, select the Check Mem program item by pressing ↓ until **Check Mem** is highlighted.
3. Select the **File Properties** command to access the Program Item Properties dialog box in Figure 10.4.
4. Specify a new name by typing **Check Disk** in the Program Title box. Press **Tab** to move to the Commands text box. Then specify the path and name of a new utility by typing **C:\dos\chkdsk.exe**.
5. Press **Enter** and Check Mem now appears as Check Disk in the Other Utils program list.
6. If you want, try running Check Disk by double-clicking on this item in the program list or by high-

lighting **Check Mem** and then pressing **Enter**. The Chkdsk utility is now run instead of the Mem utility. Press **Esc** or any other key to return to the DOS Shell.

7. Return to the Main program list by pressing ↑ until **[Main]** is highlighted. Then press **Enter**.

◆ *Lesson* ◆

11

Customizing the DOS Shell

Customizable Features

The DOS Shell includes several commands that let you customize both its appearance and operation. In this lesson, you'll learn how to use these commands to do the following:

- Change the DOS Shell display colors.
- Change the display mode from the default text mode to graphics mode.
- Create file associations, which establish links between data files and their associated programs.

Changing Colors

Here's how to change the DOS Shell display colors from the default of Ocean to one of your own choosing, such as Hot Pink:

1. Select the **Options Colors** command. You'll see the Color Scheme dialog box in Figure 11.1 which

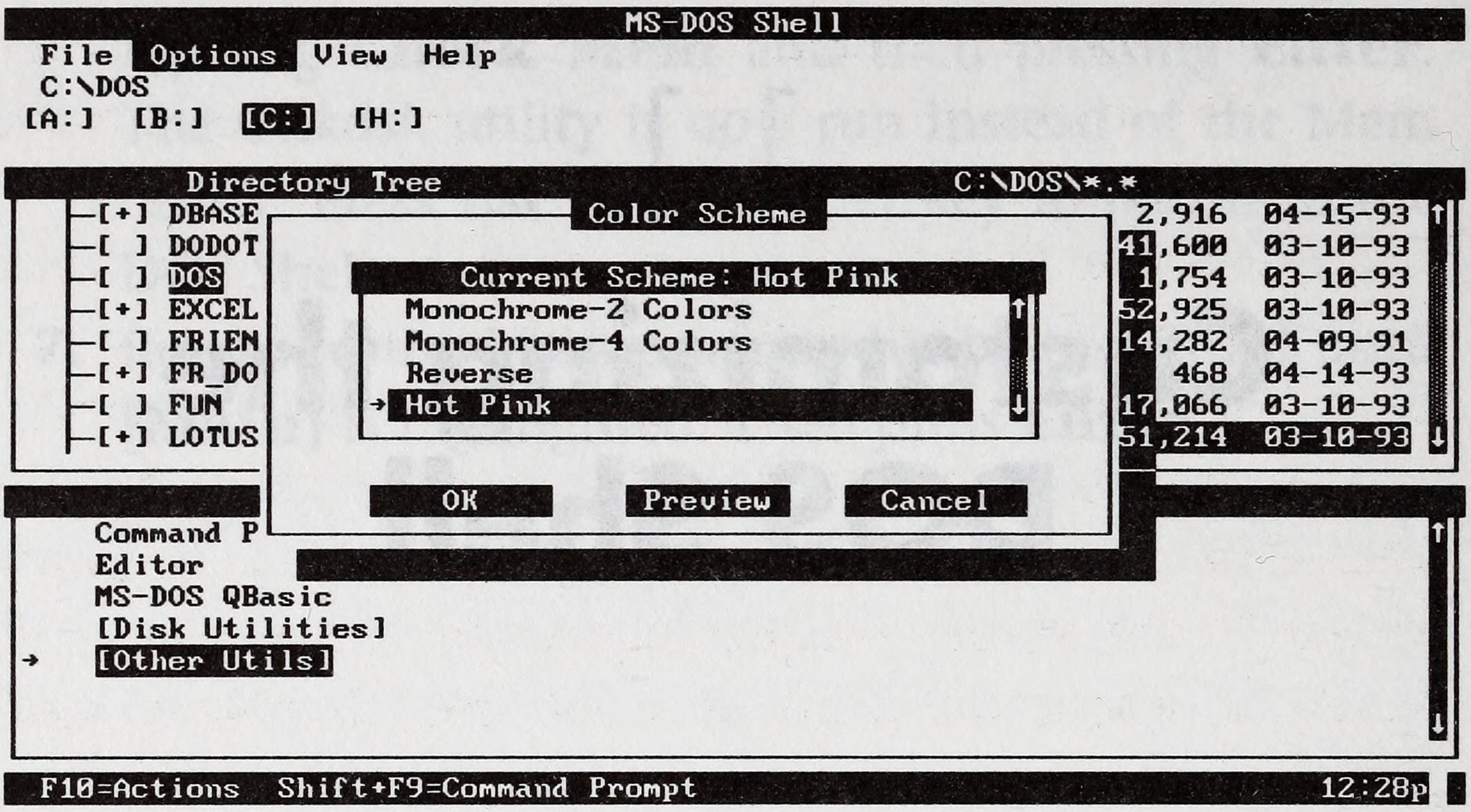


Figure 11.1 The Color Scheme dialog box

contains a list of names that correspond to different predefined color schemes.

2. Use the ↑ and ↓ keys (or the scroll bar, if you're using a mouse) to scroll through the scheme options. In this case, press ↓ 4 times to highlight **Hot Pink**.
3. If you want, press **Tab** twice to activate the **Preview** button, then press **Enter**. The DOS Shell behind the dialog box shows how your choice will look.
4. If you like your choice, press **Enter**. (If you want to select another color scheme, press **Shift+Tab** twice to return to the list.)

Changing the Display Mode

When you run the DOS Shell for the first time, its screen is displayed in *text mode*. With this setting, the DOS Shell screen can be displayed on older display adapters and

monitors. A *display adapter* is an electronic card that plugs into a slot in the back of your computer. This card, in turn, is connected to your monitor by means of a cable. Generally, the display adapter and monitor are purchased together and are of the same type—for example, Hercules, EGA, VGA, or Super VGA. Most modern display adapters and monitors, however, are also capable of displaying images in second mode known as *graphics mode*. In *graphics mode*, the shape and size of text and images on your screen appear more 3-dimensional and, therefore, more pleasing to the eye.

In both text and graphics modes, the DOS Shell will usually give you a choice of *resolution* settings—the number of lines that appear horizontally on your screen. The default setting is 25 lines, but you can usually select up to 50 lines in text mode and 60 in graphics mode. The higher the resolution you select, the more information you'll be able to see on your screen. However, both text and other objects will appear smaller if you increase the resolution.

Try changing the display mode from the default, text mode 25 lines low resolution, to the same resolution in graphics mode:

1. Select the **Options Display** command. You'll see the Screen Display Mode dialog box in Figure 11.2 containing a list of display modes. The current choice, Text 25 lines Low Resolution, is highlighted.
2. Press ↓ 3 times to highlight **Graphics 25 lines Low Resolution**.
3. If you want, press **Tab** twice to activate the **Preview** button, then press **Enter**. The DOS Shell behind the dialog box, as in Figure 11.3, shows how your choice will look.

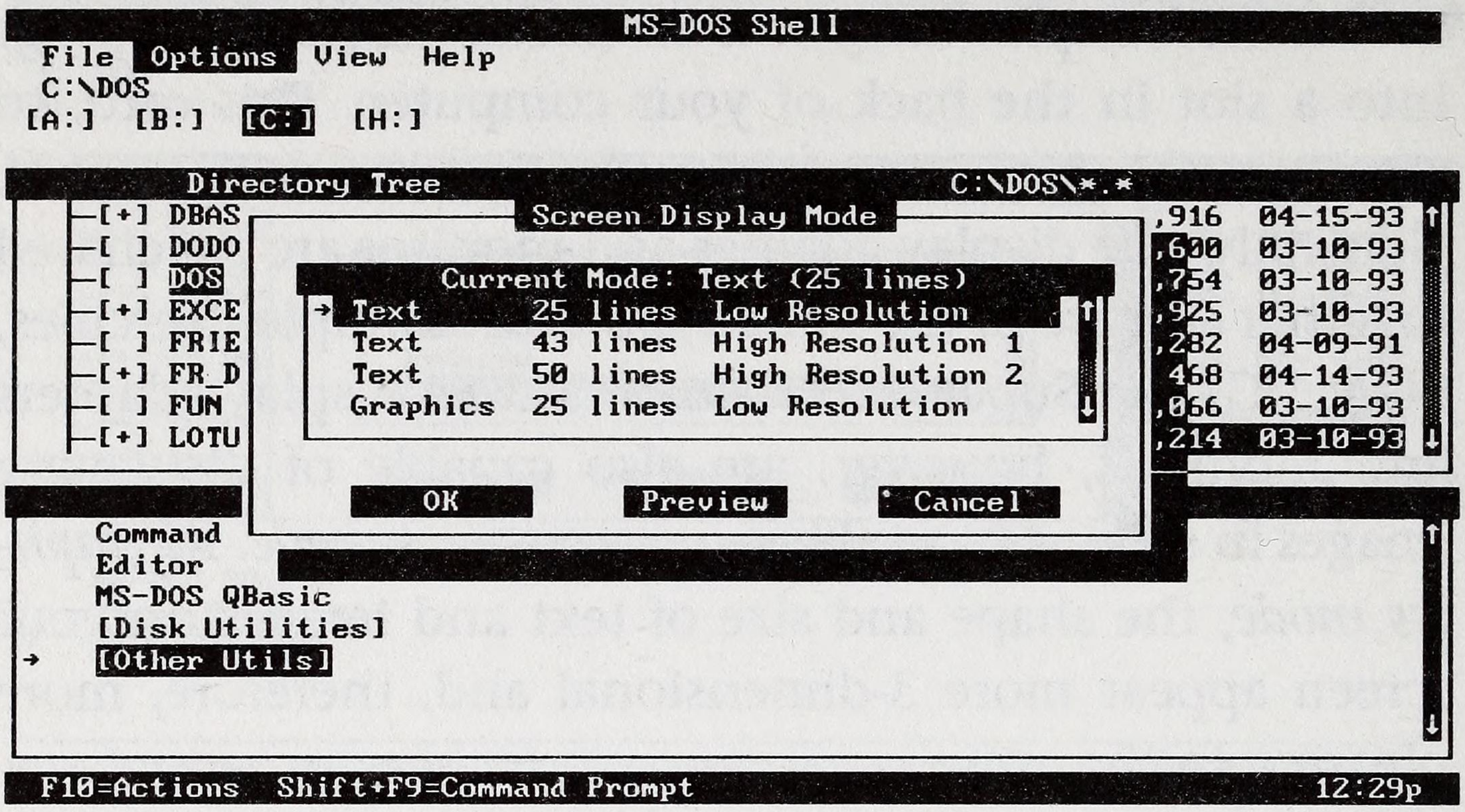


Figure 11.2 Changing the display mode

4. If you like your choice, press **Enter**. (If you want to select another display mode, press **Shift+Tab** twice to return to the list.)

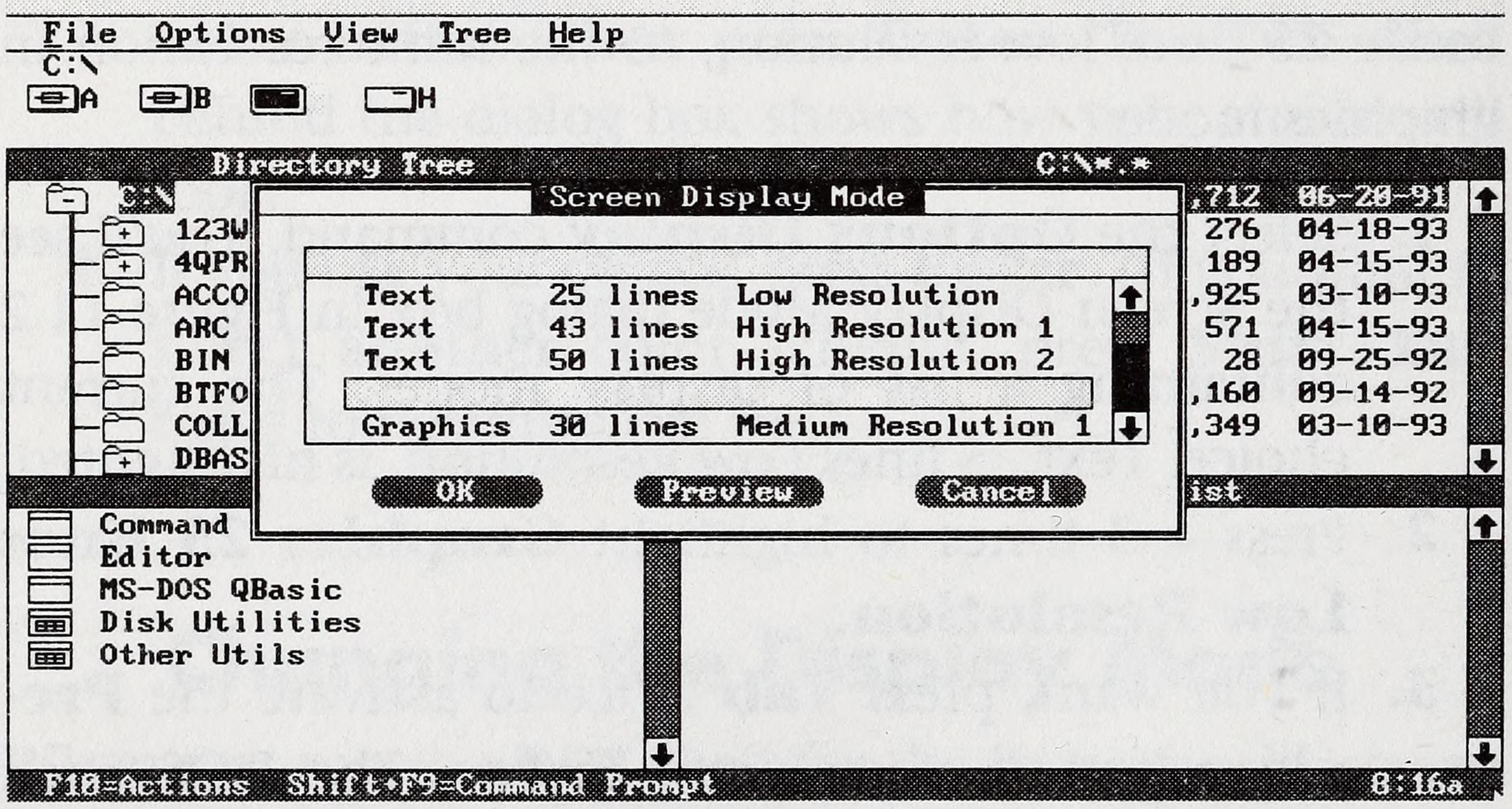


Figure 11.3 Low resolution graphics mode

Creating File Associations

In Lesson 5, you learned how to "run" files with .TXT and .BAS extensions. In the case of .TXT files, the MS-DOS Editor is started and the .TXT file you elected to run is loaded for viewing and editing. (See Lesson 13 for details on using the MS-DOS Editor.) In the case of .BAS files, the DOS Shell starts QBasic—a program that reads the instructions contained within the .BAS file and interprets them into executable code. In both cases, the DOS Shell relies on file associations.

Both .TXT and .BAS files are known as *data files*. Data files are supported—created, edited, and saved—by another host program, sometimes referred to as an *application*.

File associations create a link between data files with a specific extension—for example, .TXT—and their host application—for example, the MS-DOS Editor (C:\DOS\EDIT.COM). After a file association has been created, "running" the data file tells the DOS Shell to start the associated application and automatically load the data file into that application.

For example, let's assume you use WordPerfect for your word processing. When saving files from this application, let's also assume you give those files a .WP extension; therefore you want to create an association between WordPerfect and files with a .WP extension. Here's how to do this from the DOS Shell:

1. Press **Tab** to move to the directory tree area. Then highlight the directory that contains the application you want to associate. In this case, you want to associate .WP files with WordPerfect, so you would

press **w** until the directory containing WordPerfect, **WP51**, is selected.

2. Press **Tab** to move to the file list area. Then highlight the program that runs the application. In this case, you select **WP.EXE**, which runs WordPerfect.

Tip

For most applications, the .EXE or .COM file runs the application. See Lesson 21.

3. Create the association by selecting the **File Associate** command to access the dialog box in Figure 11.4. In the Extensions text box, type the file extension you want associated, **wp**. Notice that the initial period isn't needed.
4. Press **Enter** and the DOS Shell records the file association between WP.EXE and WP files.

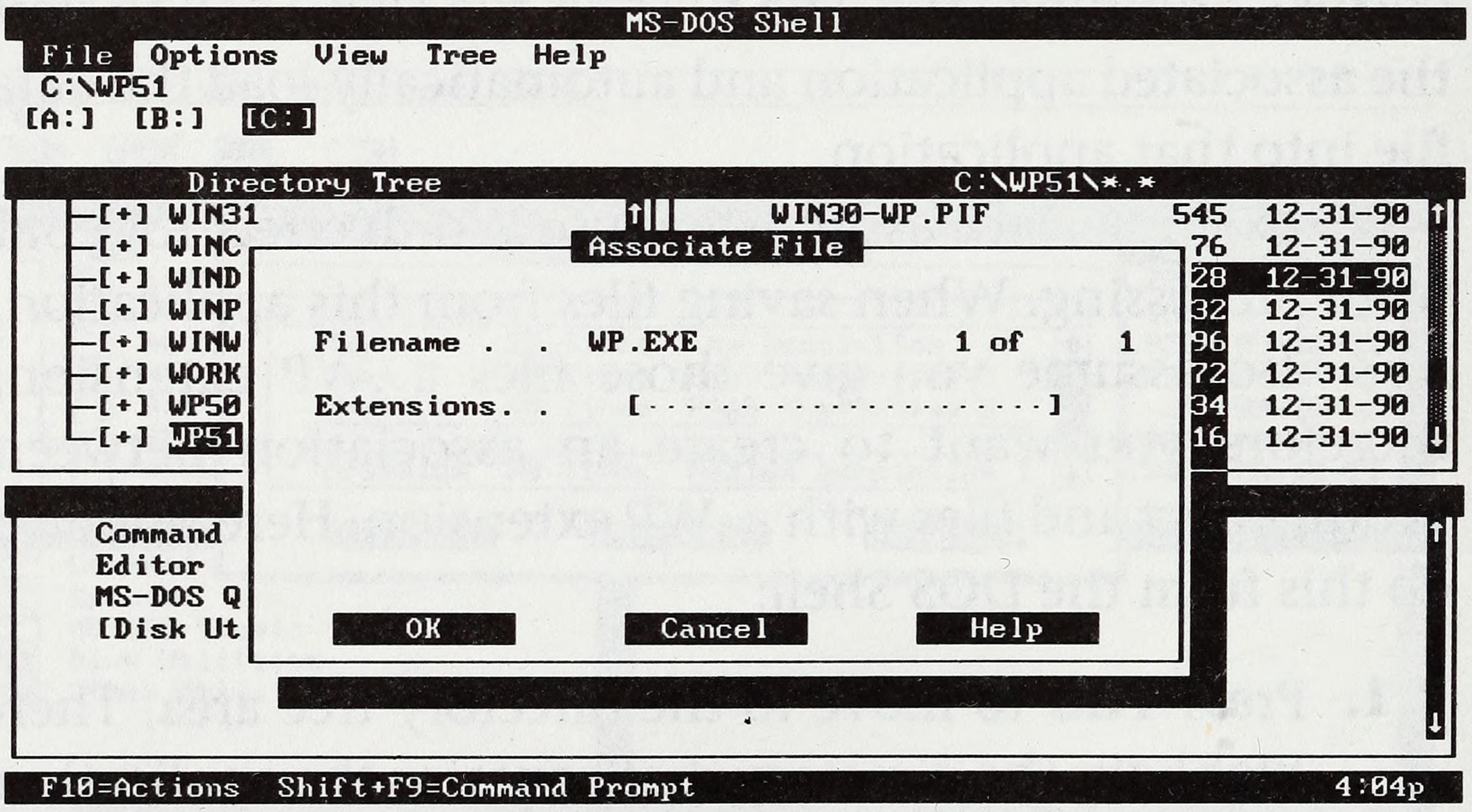


Figure 11.4 Creating a file association

Running a File Association

Once you've created a file association, you can immediately begin using it. For instance, here's how to use the DOS Shell to run the file association just created between WordPerfect and WP files:

1. Keep pressing **Tab** until the directory tree area is active. Then highlight the directory or subdirectory that contains your WP data files. For instance, you might need to select a subdirectory called **C:\WP51\WPDOC**.
2. Press **Tab** to move to the file list area. Then highlight the WP file you want to work with, **LESSON.WP** for instance, and press **Enter**. (Or double-click on Lesson.WP with the mouse.) DOS will start WordPerfect and then open LESSON.WP.
3. When you leave WordPerfect, press any key such as **Esc** to return to the DOS Shell.

◆ *Lesson* ◆

12

Special File Operations

Changing File Attributes

DOS uses attributes to control how files are accessed. A file can have one or more of the following attributes:

Archive The file appears in all file listings. This attribute is primarily used by the Backup utility (see Lesson 25) and Xcopy command (see Lesson 27) to determine which files to backup or copy. All files are assigned this attribute when they are first created.

Read only The file appears in all file listings but cannot be changed or deleted.

Hidden The file is concealed. It continues to exist but will not appear in any of the file listings.

System The file is a DOS system file, and like files with the Hidden attribute, files with this attribute do not appear in file listings.

Suppose you want to prevent another user who has access to your computer from changing or deleting the README.TXT file located in your DOS directory. You can do this by assigning a read only attribute to this file:

1. Press **Tab** until the directory tree area is active. Then keep pressing **d** until the **DOS** directory is selected.
2. Press **Tab** to move to the file list area. Then press **r** twice to select README.TXT.
3. Select the **File Change Attributes** command. You'll see the dialog box in Figure 12.1. The cursor will reside at the OK option, so press **Shift+Tab** to move back to the attribute list. Notice that an arrow indicates that README.TXT already has an archive attribute.
4. Add a Read Only attribute by clicking on **Read only** with a mouse or by pressing ↓ three times and then the **Spacebar**. An arrow to the right appears indicating that Read only is selected. If you wanted, you could now add or turn off other attributes.

⇨ ***Note:*** Remember that the Spacebar is a toggle—pressing it once turns on an attribute; pressing the Spacebar again turns off the attribute.

5. Press **Tab** to move to OK and then press **Enter**.

Once the Read only attribute has been assigned to a file, you can't change the contents or delete the file. Instead,

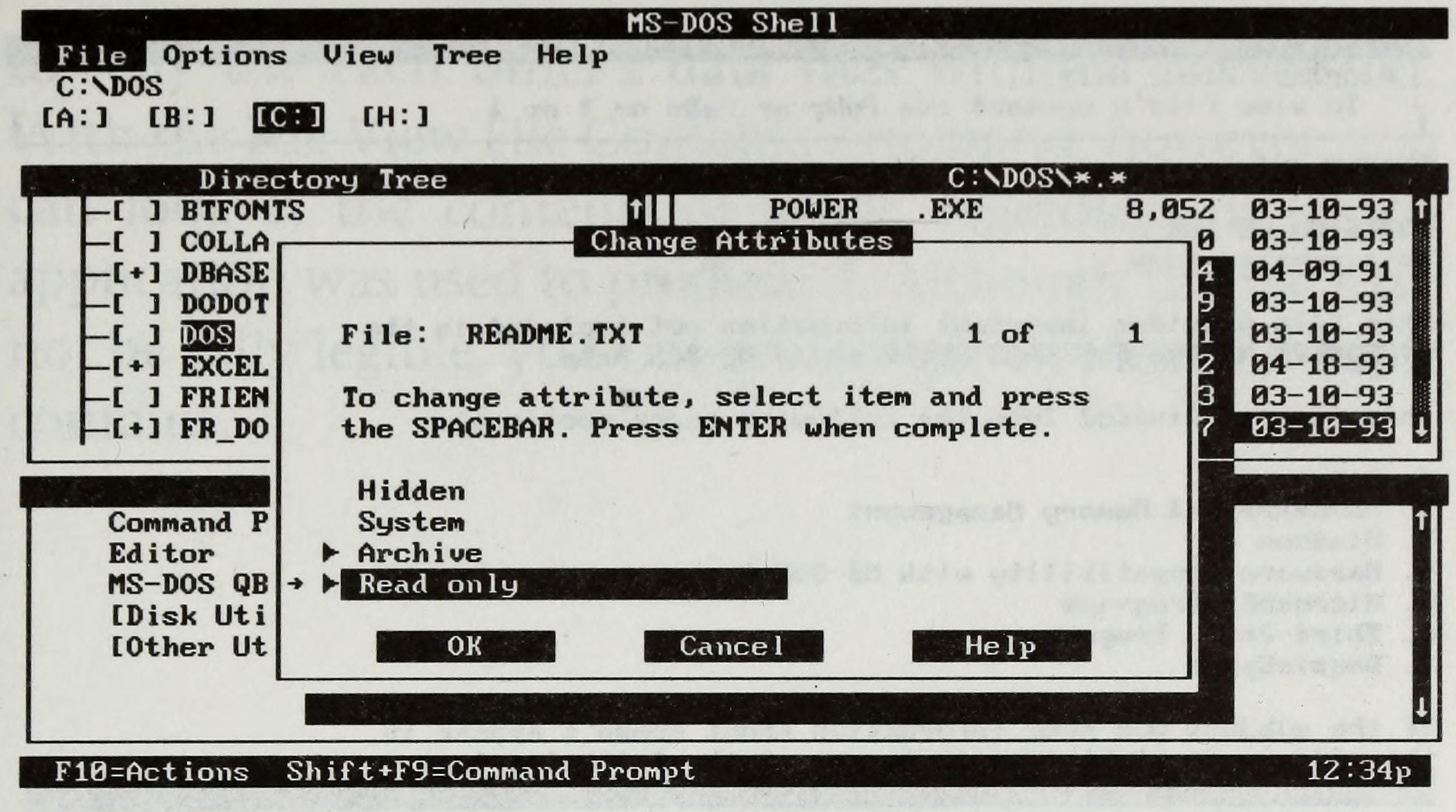


Figure 12.1 Changing file attributes

you must first turn off the Read only attribute by repeating the above procedure.

⇨ ***Note:*** You can also change the attributes of a group of files. To select a group of files, see Lesson 6. When you use the File Change Attributes command, you'll be able to change the attributes for the entire group or for each file individually.

Viewing a File's Contents

The DOS Shell provides an easy way to view the contents of any file. For instance, try viewing the contents of README.TXT located in your DOS directory:

1. If you need to, keep pressing **Tab** to make the directory tree area active, then press **d** until the **DOS** directory is selected. Press **Tab** to move to the file list area, then press **r** twice to highlight **README.TXT**.

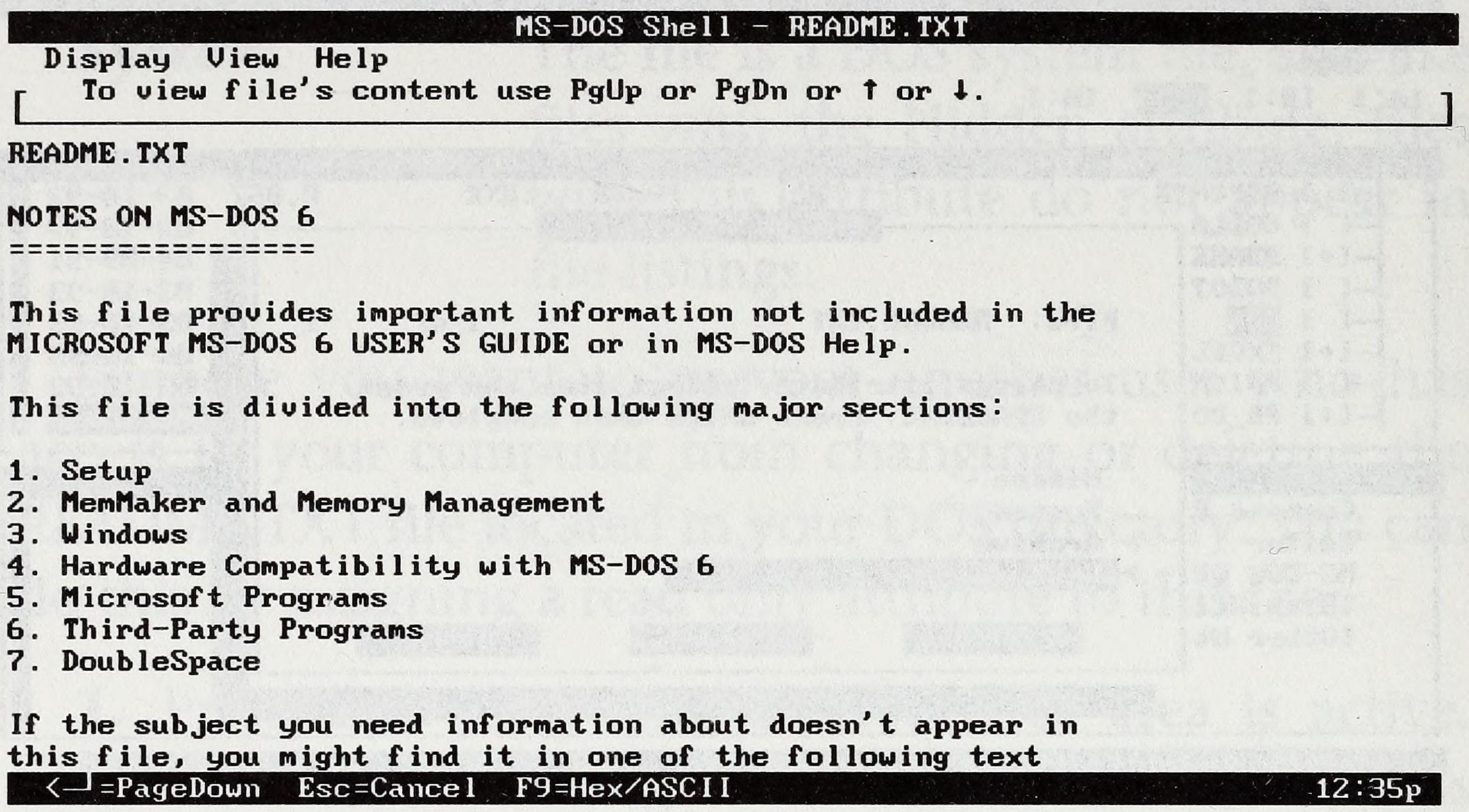


Figure 12.2 Viewing the contents of a file

2. Press **F9**, the shortcut for the **File View File Contents** command. As you can see in Figure 12.2, you'll see README.TXT in a viewer window.
3. To scroll the file up or down, press **PgUp**, **PgDn**, ↑, or ↓.
4. To return to the DOS Shell, press **Esc**.

The File View File Contents command offers multiple advantages over other ways of viewing a file. To begin with, it's quick; if you can't remember what a file contains, just use this command to have a look.

Another advantage is that you can view just about any file, regardless of its format. As you begin to run applications under DOS, you'll discover that different applications save their data files in different formats. This means that two word processing programs, for example, cannot

directly use each other's data files without conversion. With the File View File Contents command, however, you can look at the contents of a file, regardless of which application was used to produce it. Although the file may not be fully legible, you can usually get an idea of what it contains.

◆ *Lesson* ◆

13

Using the MS-DOS Editor

Editing ASCII Files

As noted earlier in the book, ASCII is an acronym for American Standard Code for Information Interchange, which is a commonly accepted file format for text files. This format is supported by numerous DOS applications; in fact, most popular DOS word processing, spreadsheet, and database packages are capable of reading and writing files in ASCII format. For this reason, ASCII files are often used as a means by which to share information between DOS applications.

Both DOS 5 and 6 come with their own ASCII file editor, MS-DOS Editor. You can use this program to create, edit, and print simple letters and memos, and to create special files that automate DOS operations. In Lesson 29, for example, you'll learn how to use the MS-DOS Editor to create batch files (files with a .BAT extension containing DOS commands). You can run a .BAT file just as you would any other program to execute the commands it contains.

Creating a Memo

The MS-DOS Editor is a snap to use. As an exercise, let's assume you have to dash off a quick memo to one of your associates down the hall. Here's how to do this:

1. Start the MS-DOS Editor by pressing **Tab** until the Main program list area is active. Press ↓ so that **Editor** is highlighted and then **Enter**. (Using a mouse, just double-click on Editor.) You'll see the dialog box in Figure 13.1 asking you for an ASCII file to load into the Editor.
2. To create a new memo (file), press **Enter** to bypass this dialog box. You'll now see the opening screen of the MS-DOS Editor.
3. Press **Esc** to clear the opening prompt from the screen.

⇨ ***Note:*** To view the MS-DOS Editor Help, press Enter instead. While you're working in the Editor, pressing F1 at any time also accesses this Help.

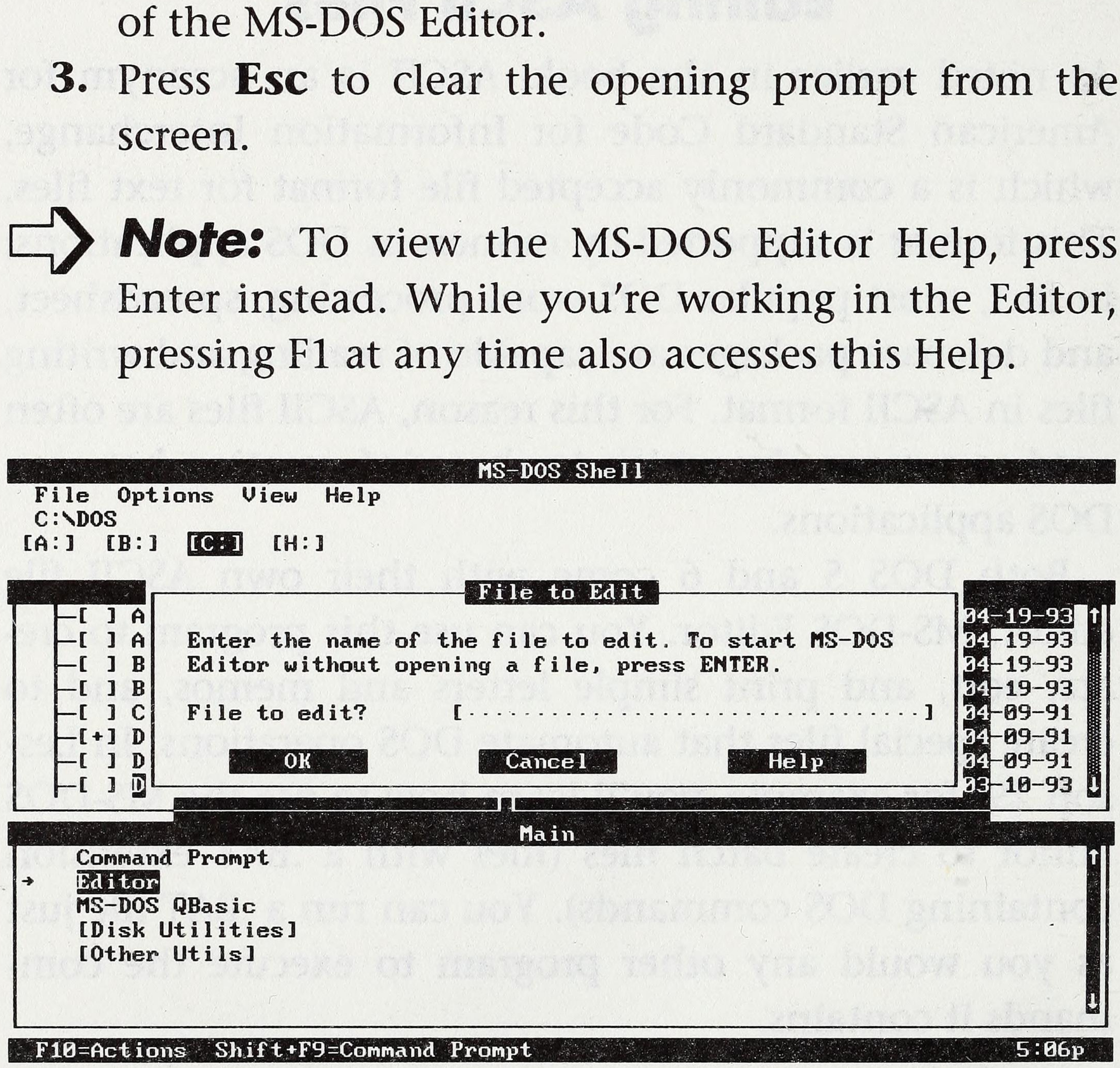


Figure 13.1 Opening the Editor

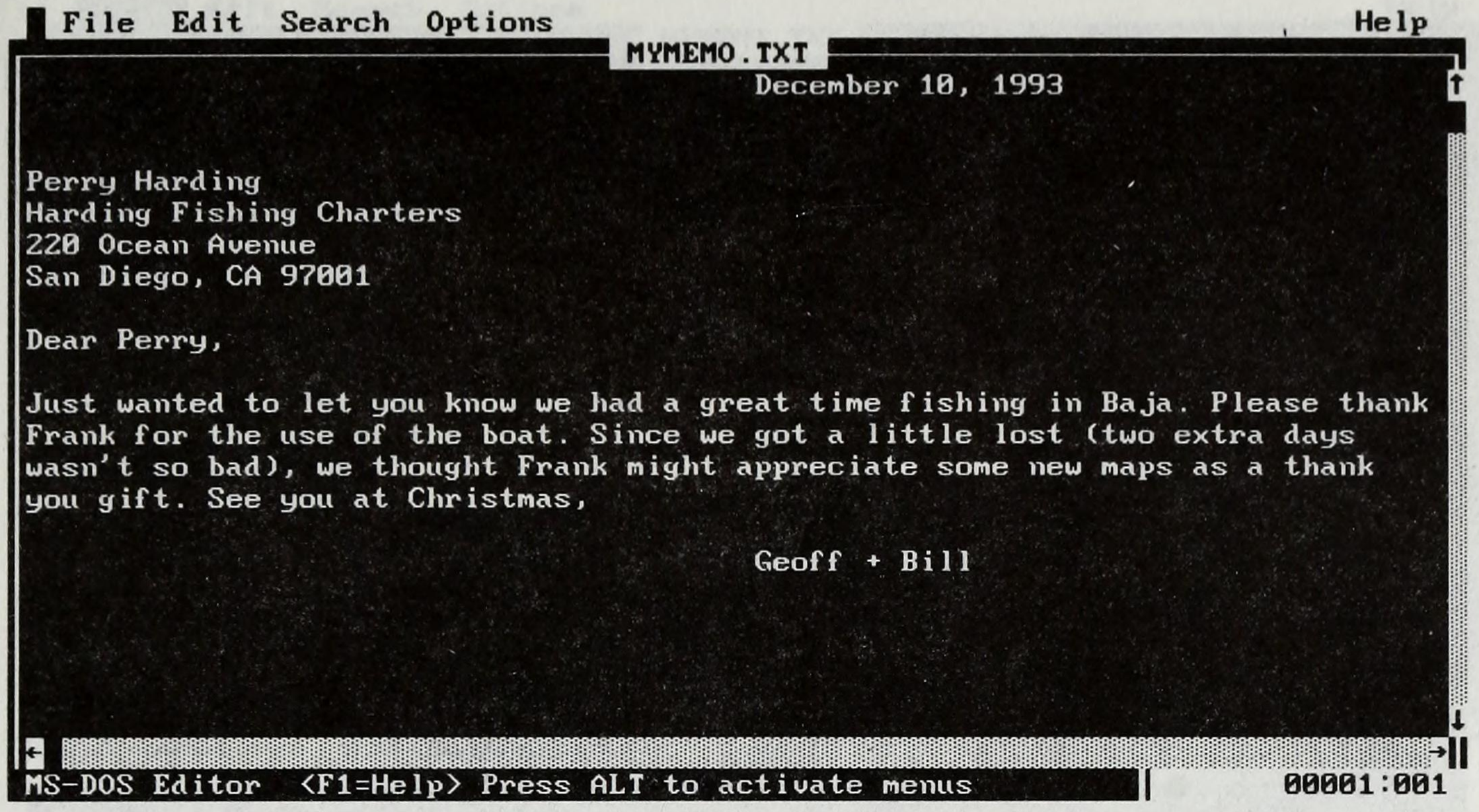


Figure 13.2 A sample memo in the Editor

4. Begin to type a memo, such as the one shown in Figure 13.2. When you get to the end of a line, press **Enter.** (There's no automatic wordwrap in the Editor program.)

Tip

To print a file you've created in the Editor, you'll want to keep each line of text no longer than the width of the screen.

Saving the Memo

Once you've created your text, you're ready to save it to disk. For example, try saving the memo shown in Figure 13.2 under the name MYMEMO.TXT:

1. Select the **File Save As** command from the Editor menu to get the dialog box shown in Figure 13.3.
2. Type the name you want to save the file under in the File Name text box. In this case, just type

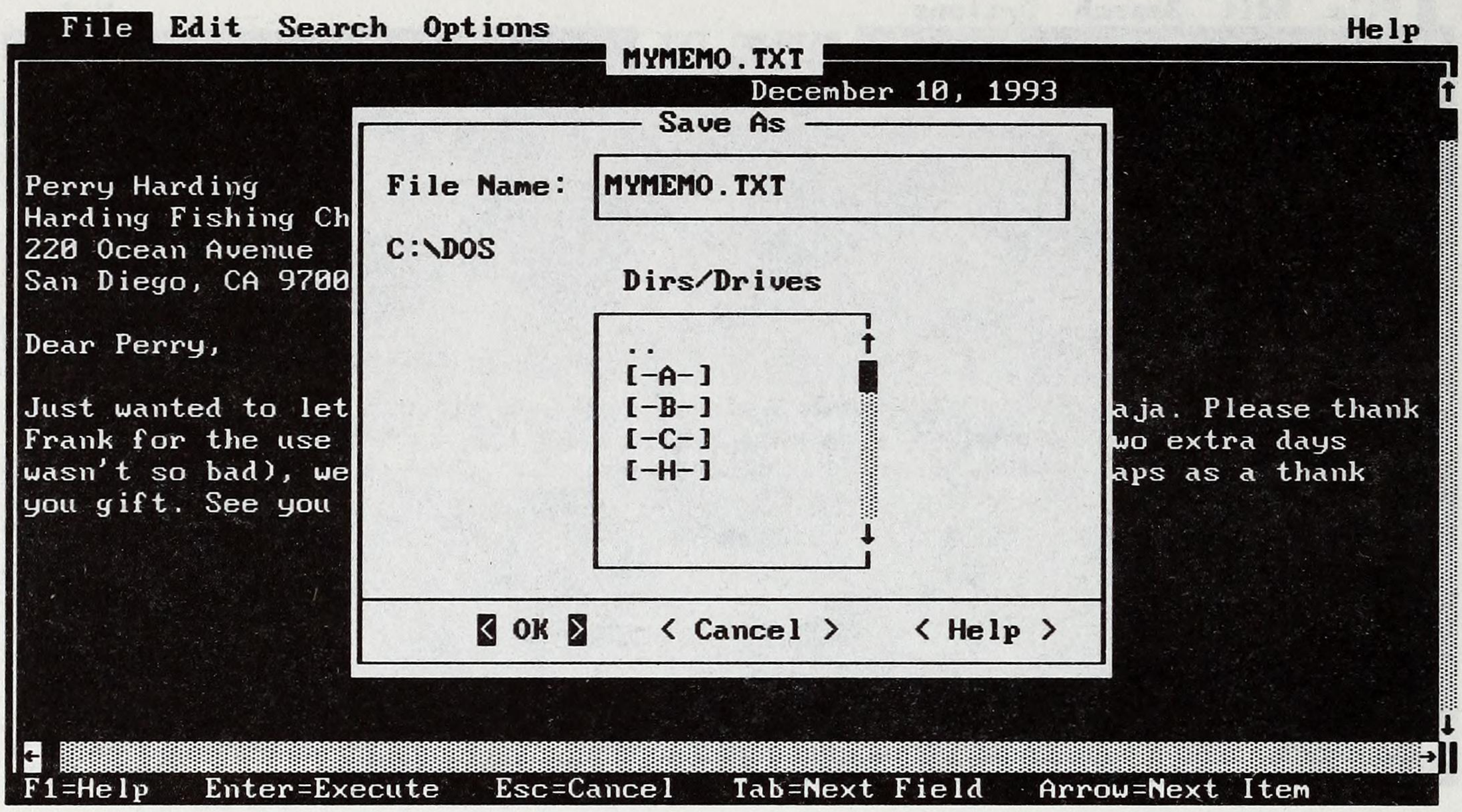


Figure 13.3 Saving a file

mymemo.txt and press **Enter**. DOS saves this file MYMEMO.TXT in the DOS directory and now uses this name at the top of Editor.

⇨ ***Note:*** If you don't specify a path, DOS saves the file in the current directory shown to the left, in this case C:\DOS. You can specify a different path by typing it in front of the filename or by selecting a directory from the Dirs/Drive list.

After you've saved a file once, you can thereafter use the File Save command.

Printing the Memo

You can print the file currently open directly from the MS-DOS Editor. For example, try printing the memo we've been working with in the file MYMEMO:

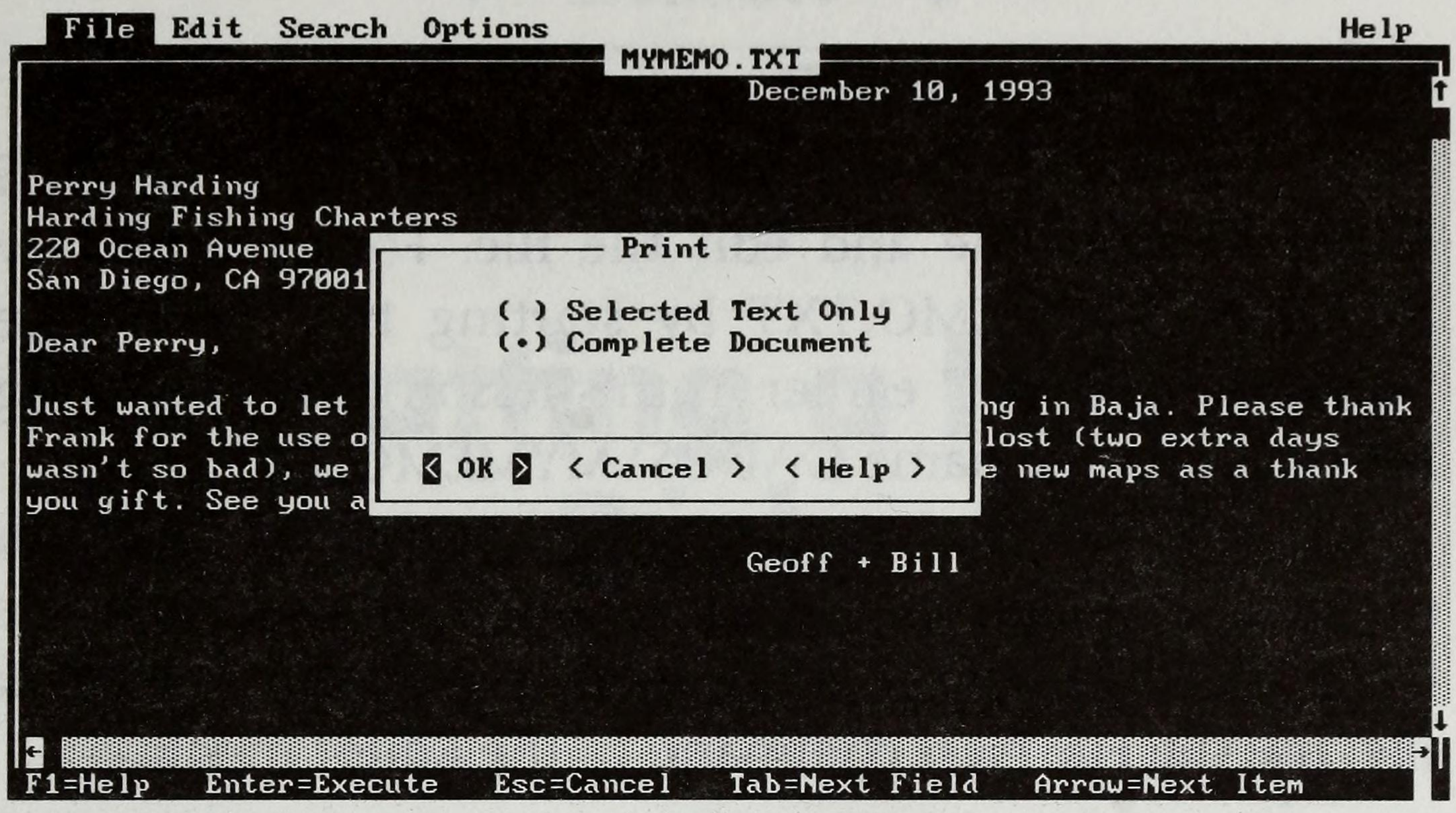


Figure 13.4 Printing a file

1. Make sure that your printer is turned on and connected to the first parallel port LPT1. (Otherwise, you won't be able to print from the Editor.)
2. Select the **File Print** command from the Editor menu. You'll see the dialog box shown in Figure 13.4.
3. Since **Complete Document** is already selected, press **Enter** to print the document.

Exiting the MS-DOS Editor

To leave the MS-DOS Editor and return to the DOS Shell, select the **File Exit** command from the Editor menu. If you've neglected to save your changes to the current file before trying to exit, Editor will notify you of this and give you a chance to save your changes before exiting.

Editing the Memo

Once you've saved a text file, you can start the MS-DOS Editor at any time and edit the file. For example, try reopening MYMEMO.TXT by starting the Editor (See "Creating a Memo" earlier in this lesson) and specifying the path and file name C:\DOS\MYMEMO.TXT.

When you're working in the Editor, you can open an existing .TXT file such as MYMEMO.TXT using the File Open command in the Editor menu. To open a new blank file, use the File New command in the Editor menu.

◆ *Lesson* ◆

14

Formatting a Floppy Disk

Why Format?

Before you can use a floppy disk—that is, create directories on it or copy files to it—the disk must be formatted. When a disk is *formatted,* it contains a root directory and a *file allocation table*—a special type of file that tells DOS where to look on the disk for files and directories.

Testing for Formatting

From the DOS Shell you can test whether a disk in a floppy drive has been formatted. For example, follow these steps to test whether a floppy disk has been formatted:

1. Insert a floppy disk in drive A or B of your PC.
2. Press **Tab** until the drive icons area of the DOS Shell is active. Then use → or ← to highlight the appropriate drive—for example, A:—and press **Enter**. (Using a mouse, just double-click on [A:].)

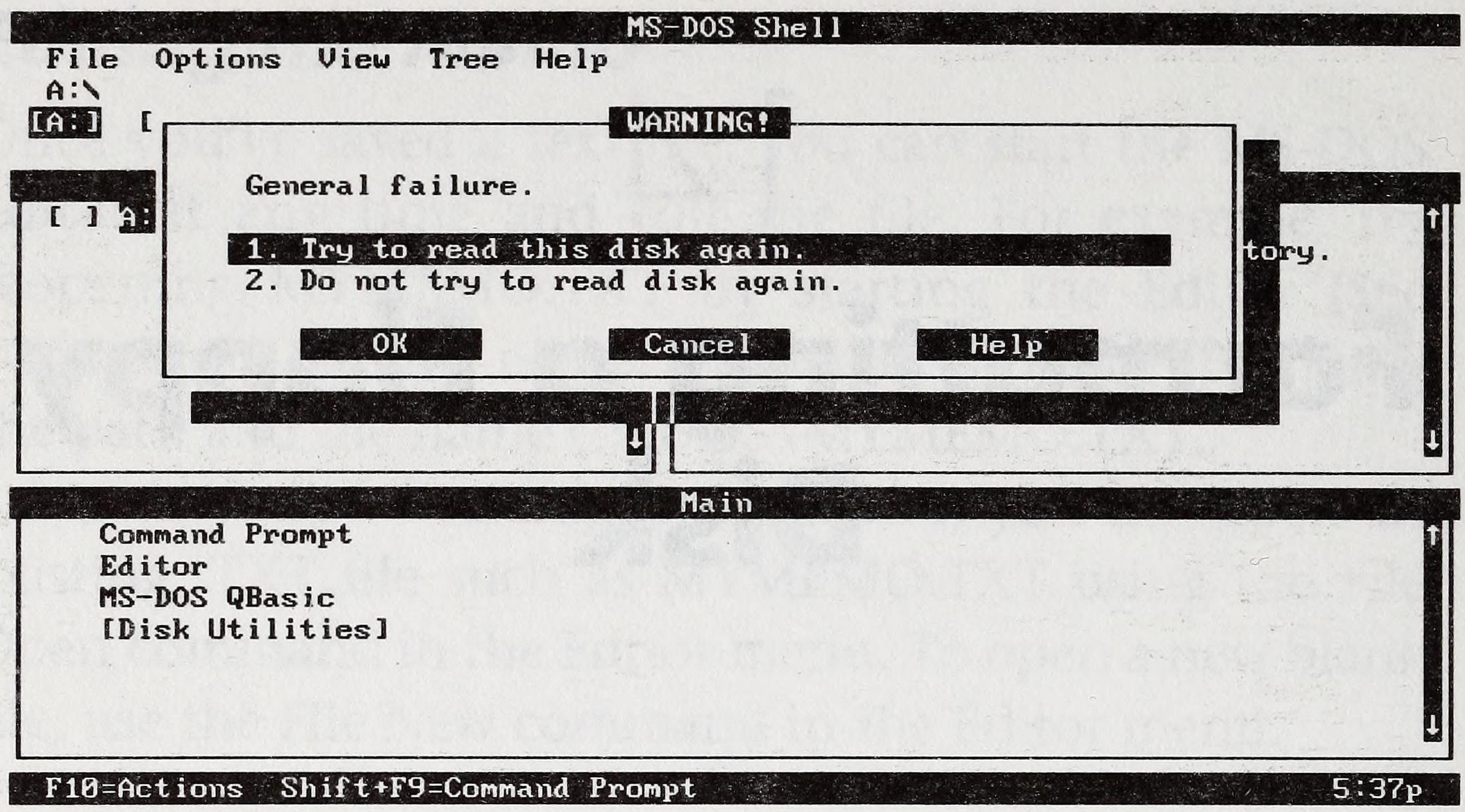


Figure 14.1 A General Failure message

3. If the disk has been formatted, the directories and files on the disk will be displayed; otherwise you'll see the General Failure message shown in Figure 14.1, indicating the disk needs formatting (or that the disk has not been properly placed in the drive). Press **Esc** to clear the message.

Formatting a New Disk

From the DOS Shell, here's how to format one or more disks:

1. Insert a disk into a floppy drive (usually drive A for 5.25 inch disks or drive B for 3.5 inch disks.)
2. Keep pressing **Tab** until the Main program list area is active. Press ↓ three times and then **Enter** to choose the **[Disk Utilities]** group. In the Disk Utilities program list that appears, select **Format** by pressing ↓ five times and then **Enter**. You'll see

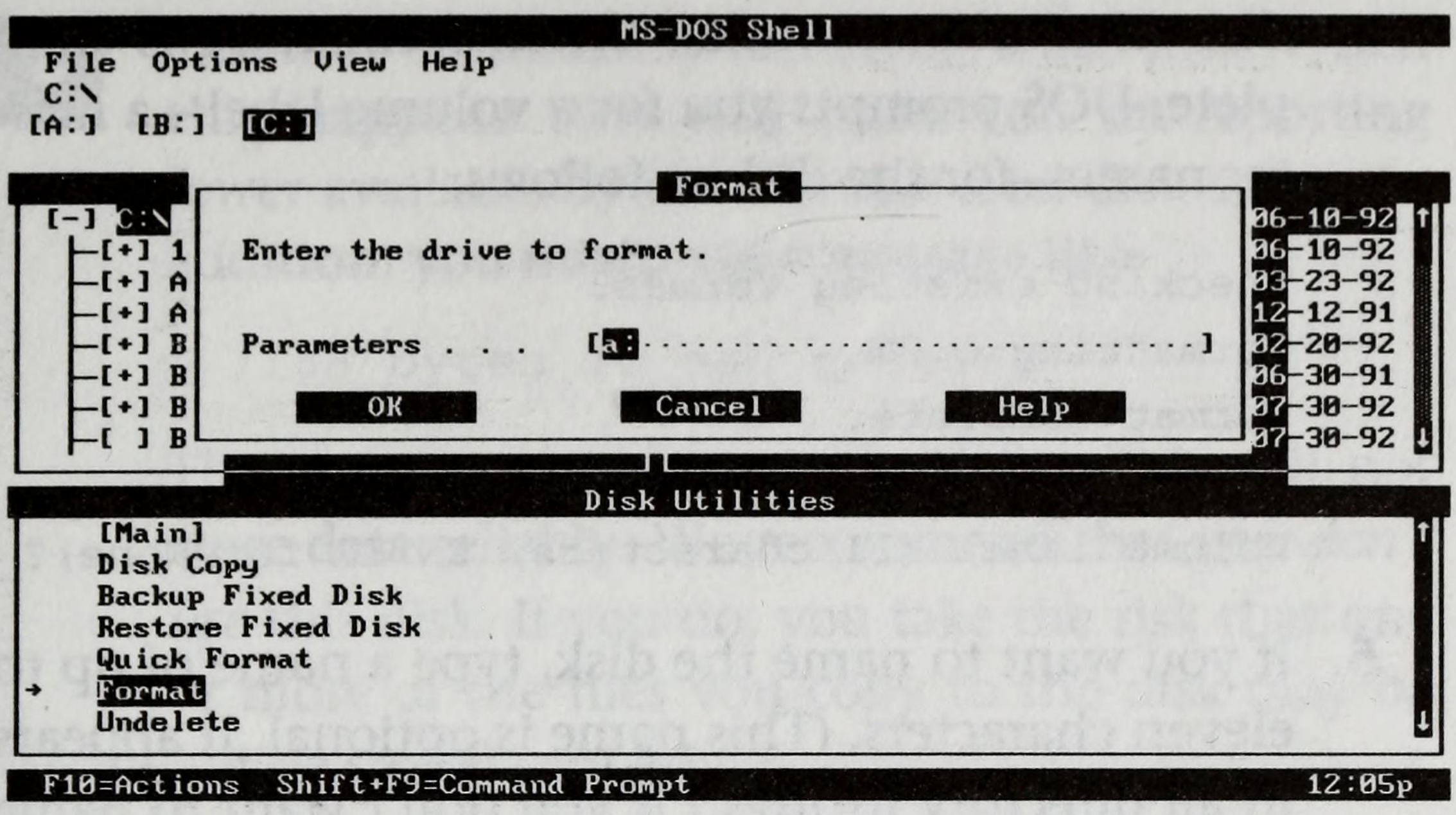


Figure 14.2 Formatting a floppy disk

the dialog box in Figure 14.2 suggesting drive A (a:) as the drive to format.

Formatting erases all the files and directories that are already on a floppy disk. Therefore, make sure the disk you are about to format is either completely blank or does not contain files you want to save.

3. Type the letter of the drive containing the disk you want to format followed by a colon—for example **a:**—and press **Enter**. Your screen will go blank, and you'll see the following instructions displayed:

```
Insert new diskette for drive A
and press ENTER when ready...
```

4. Press **Enter** to proceed with formatting. DOS reports its percentage of completion as it formats the

disk (this takes a while). When formatting is complete, DOS prompts you for a volume label—a master name—for the disk as follows:

```
Checking existing format.
Formatting 1.2M
Format complete.

Volume label (11 characters, ENTER for None)?_
```

5. If you want to name the disk, type a name of up to eleven characters. (This name is optional. It appears in all directory listings.) If you don't want to name the disk, just press **Enter** to continue. Either way, DOS reports the total space it found on the disk and asks you whether you wish to format another:

```
1213952 bytes total space
1213952 bytes available on disk

        512 bytes in allocation unit.
        2371 allocation units available on disk.

Volume Serial Number is 357E-13D0

Format another (Y/N)?
```

6. If you have another disk you want to format, type **Y** and press **Enter**; DOS will take you through another formatting cycle; otherwise, type **N** and press **Enter**. DOS displays the following:

```
Press any key to return to MS-DOS Shell....
```

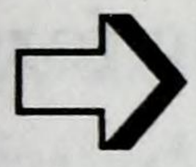
7. Press any key. The DOS Shell is once again displayed on your screen.

You may occasionally run across a bad disk. When this happens DOS will warn you by reporting fewer available bytes than the total disk space. In addition, you might see a message like

```
7168 bytes in bad sectors
```

This means a small portion of the disk will not store data reliably. We recommend that you *don't use* this disk. If you do, you take the risk that one or more of the files you copy to the disk may be not be stored properly.

8. Return to the Main program group by pressing ↓ five times to highlight **[Main]** and then **Enter**.



Note: You can also format disks from the DOS command line prompt using the FORMAT command. See Lesson 21.

Performing a Quick Format

DOS's convenient Quick Format feature, first introduced in DOS 5, lets you quickly erase all the directories and files on a floppy disk. This feature is handy when you want to reuse a floppy disk full of old, outdated information that you no longer need.

Try quick formatting a disk containing old, unneeded files of your own by following the procedure for formatting a new disk except select **Quick Format** in the Disk Utilities program list. You'll see the same messages and instructions from DOS except Quick Format will appear instead of Format. For example, when you select Quick

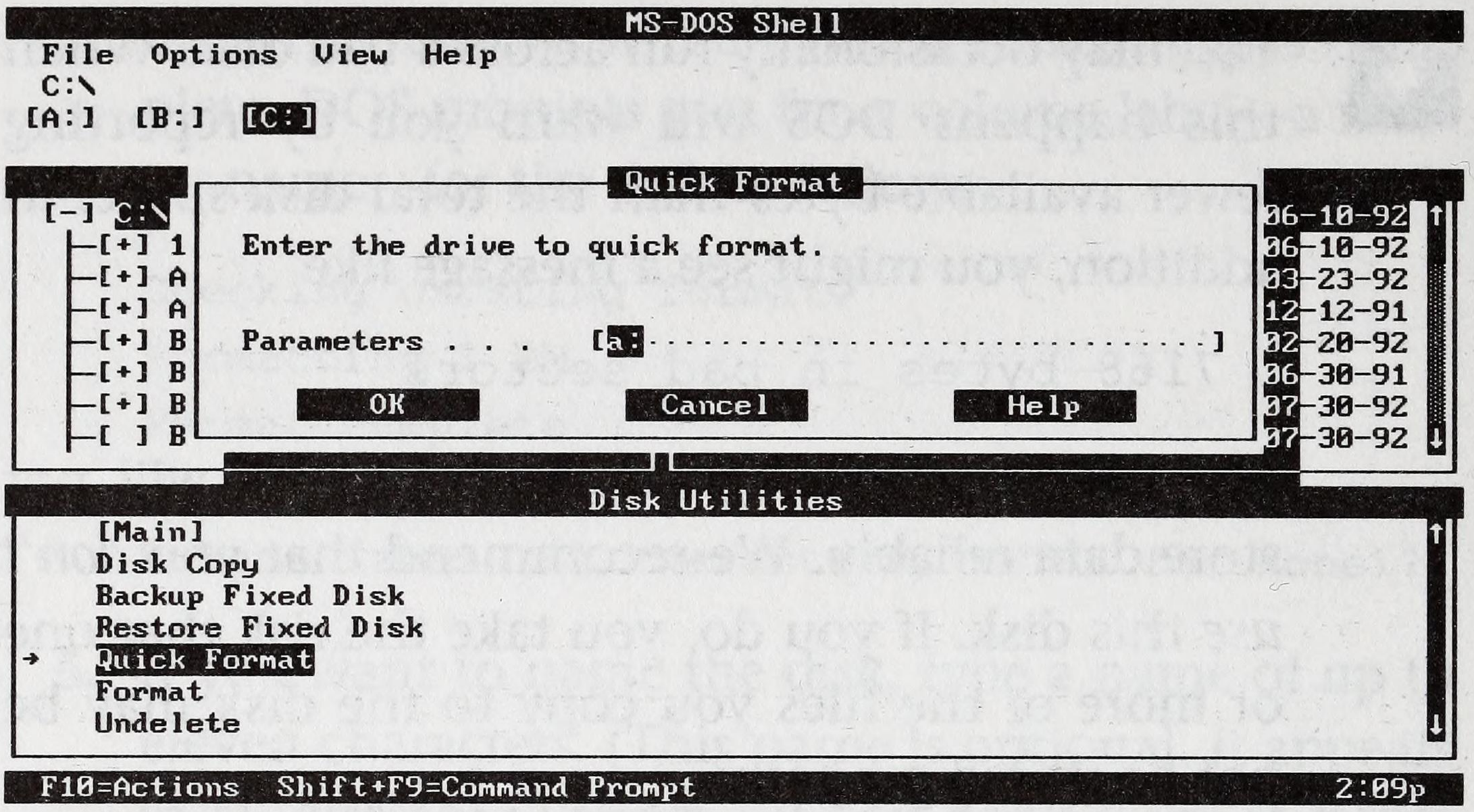


Figure 14.3 Quick formatting a floppy disk

Format, you'll see the dialog box in Figure 14.3 asking for the drive to quick format.

⇨ ***Note:*** If you try to quick format an unformatted floppy disk, you'll be prompted to format the disk instead.

Creating a System Disk

Most PCs boot from an internal hard-disk drive. To do this, they read the DOS file COMMAND.COM and two other hidden system files stored on the root directory of the hard disk (C:\). This allows a PC to boot quickly and efficiently. The one drawback, though, is that you may not be able to start your PC in the event your hard disk fails. For this reason, experienced users always keep a *system disk* (a bootable floppy disk) handy. That way, they can at least start the computer from a floppy disk drive

and possibly recover some of the data on the failed hard disk.

You can create a system disk through the DOS format utility by following the procedure for formatting a new disk (See "Formatting a New Disk" earlier in this lesson) with one exception. When DOS prompts you for a drive to format in step 3, type the target drive followed by a space then / (slash) then s, as in **a: /s**.

During the formatting session, you'll see the message "System Transferred," which indicates that a system disk has been created. Don't forget to put this disk in a safe place.

If your hard disk ever fails, you can place the system disk in drive A (or B) and then restart your PC. This tells your PC to boot from the system disk rather than the failed hard disk.

◆ *Lesson* ◆

15

Copying Floppy Disks

Understanding Disk Copy

The Disk Copy utility (DISKCOPY.EXE) lets you create an exact duplicate of a floppy disk. Disk Copy is often used to make backup copies of important floppy disks. Most software vendors recommend, for example, that you make backup copies of the disks on which their program is shipped in case the originals are ever damaged. The Disk Copy utility is ideally suited for this purpose.

Because the Disk Copy utility makes an *exact*—track for track—copy of a floppy disk, it can only be used to copy information between two disks of the same type (same size and amount of space available). This means you can't use Disk Copy to copy information between a 5.25 inch, 1.2 Mb disk and a 3.5 inch, 1.44 Mb disk, for instance.

Although the Disk Copy utility does not affect the source disk (the disk that is being copied), it does erase all data on the destination disk (the disk that is receiving information). Therefore, make absolutely sure the destination disk is empty or does not contain any information

you want to keep. If the destination disk is unformatted, DOS will format it for you while it is carrying out the Disk Copy procedure.

Using Disk Copy

For this exercise, let's assume you want to make backup copies of your DOS 6 distribution disks. Let's also assume you received the program on three 3.5 inch, high-density (1.44MB) disks. Therefore, to create backups, you'll need to have three blank, 3.5 inch, high-density disks at hand. When you're ready, perform the following steps:

1. Insert the first DOS disk (labeled **Disk 1-Setup**) into Drive B (your 3.5 inch drive.)
2. Keep pressing **Tab** until the Main program list area is active. Press ↓ three times and then **Enter** to choose the **[Disk Utilities]** group. In the Disk Utilities program list that appears, select **Disk Copy** by pressing ↓ and then **Enter**. You'll see the dialog box in Figure 15.1 suggesting drive A (a:) as the drive to copy from (source) and drive B as the drive to copy to (destination).
3. Using the keyboard, change the source and destination parameters that read **a: b:** to read **b: b:**. When you're done, press **Enter**. Your screen will go blank and then display the following message:

```
Insert SOURCE diskette into drive B:
Press any key to continue. . .
```

4. Press **Enter** in response to this prompt. DOS displays a status message and begins copying the infor-

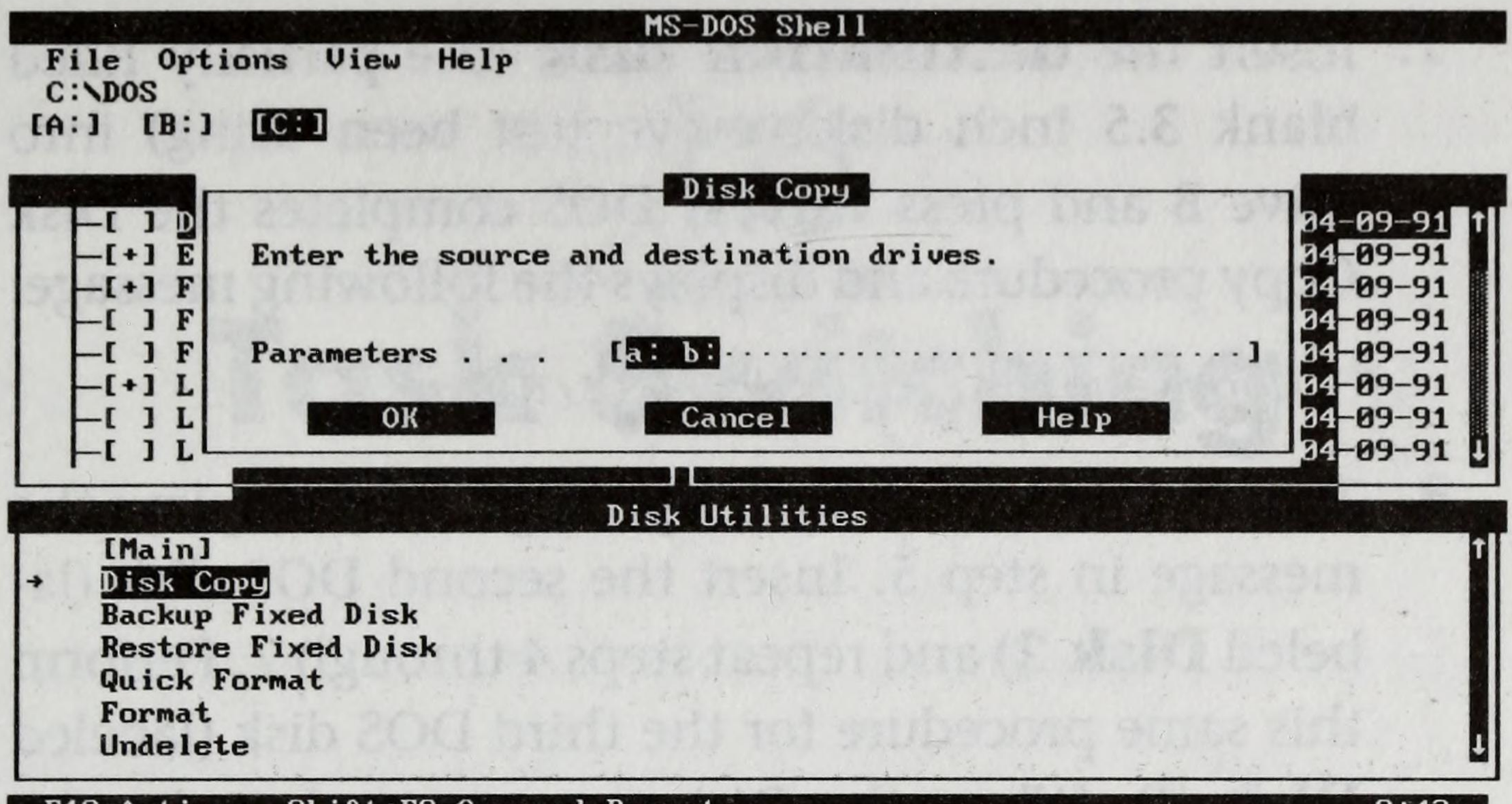


Figure 15.1 Using the Disk Copy utility

mation on the source disk into your computer's RAM. When RAM is full, DOS displays the message:

```
Insert TARGET diskette into drive B:
Press any key to continue. . .
```

5. Insert the **destination disk** (one of your blank 3.5 inch disks) into drive B and press **Enter**. DOS copies the information in RAM onto the destination disk. When this is done, DOS displays the message:

```
Insert SOURCE diskette into drive B:
Press any key to continue. . .
```

6. Insert the **source disk** (labeled **Disk 1**) into drive B and press **Enter**. DOS copies the rest of the information on the disk into RAM. When this is done, DOS displays the message:

```
Insert TARGET diskette into drive B:
Press any key to continue. . .
```

7. Insert the **destination disk** (the partially filled blank 3.5 inch disk you've just been using) into drive B and press **Enter**. DOS completes the Disk Copy procedure and displays the following message:

```
Copy another diskette (Y/N)?
```

8. Type **Y** in response to this prompt. DOS displays the message in step 5. Insert the second DOS disk (labeled **Disk 2**) and repeat steps 4 through 7. Perform this same procedure for the third DOS disk (labeled **Disk 3**). When the Disk Copy procedure for the third source disk is complete, Type **N** in response to the message in step 7. DOS displays the message:

```
Press any key to return to MS-DOS Shell....
```

9. Press any key. The DOS Shell screen is once again displayed.

◆ *Lesson* ◆

16

Task Switching

About the Task Swapper

With the aid of the DOS Shell Task Swapper, you can run multiple programs at the same time and switch back and forth between them. You might use this capability, for example, to compose a business letter in a word processing program while at the same time crunching the numbers for the letter in a spreadsheet program.

Turning the Task Swapper On and Off

Before you can start multiple programs, you must turn on the Task Swapper; it's turned off by default. Here's how to do this from the DOS Shell:

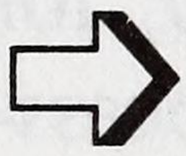
1. Select the **Options Enable Task Swapper** command.
2. To see that the Task Swapper is turned on, select the **Options** command. Notice that a rectangular dot now appears to the left of Enable Task Swapper

indicating that the Task Swapper is turned on. Notice also that a new area is added to the lower right corner of the DOS Shell screen called the Active Task List. Press **Esc** to return to the DOS Shell.

Whenever you want to turn off the Task Swapper, simply select the Options Enable Task Swapper command again.

Starting More Than One Program

In this section we'll show you how to simultaneously run the programs mentioned above—a word processing program and a spreadsheet program. For the sake of simplicity, we'll use the MS-DOS Editor (see Lesson 13) for the word processing program and Quattro Pro for DOS from Borland International for the spreadsheet program.



Note: To run a program from the DOS Shell that is not included in the program list area, you'll need to know the name of its main program (executable) file. Usually, an executable file has an .EXE or a .COM extension. For example, the main program file for Quattro Pro for DOS is Q.EXE and is found in the C:\QPRO directory.

Here's how to simultaneously run the Editor and Quattro Pro for DOS or two programs of your own choosing:

1. Make sure that the Task Swapper is enabled.
2. Open the first program, the MS-DOS Editor, by pressing **Tab** until the Main program list area is active. Press ↓ and then **Enter** to select **Editor**. Press **Enter** again to run the Editor. (If you're using

a mouse, simply double-click on Editor.) When the Editor appears on your screen, press **Esc** to clear the MS-DOS Editor dialog box.

3. Press **Ctrl+Esc** to return to the DOS Shell. Notice that Editor appears in the Active Task List shown in Figure 16.1.
4. Open the second program—Quattro Pro for DOS in this example—by pressing **Tab** three times so that the directory tree area is active. Press **Home** to move to the root directory **C:**, and then keep pressing **q** until the **QPRO** directory is highlighted. Press **Tab** to move to the file list area and then keep pressing **q** until **Q.EXE** is highlighted. Press **Enter** to run Quattro Pro for DOS.
5. When the second program, Quattro Pro for DOS, appears on your screen, press **Ctrl+Esc** to return to the DOS Shell. Notice that the name of Quattro Pro's executable file Q.EXE also appears in the active task list shown in Figure 16.1.

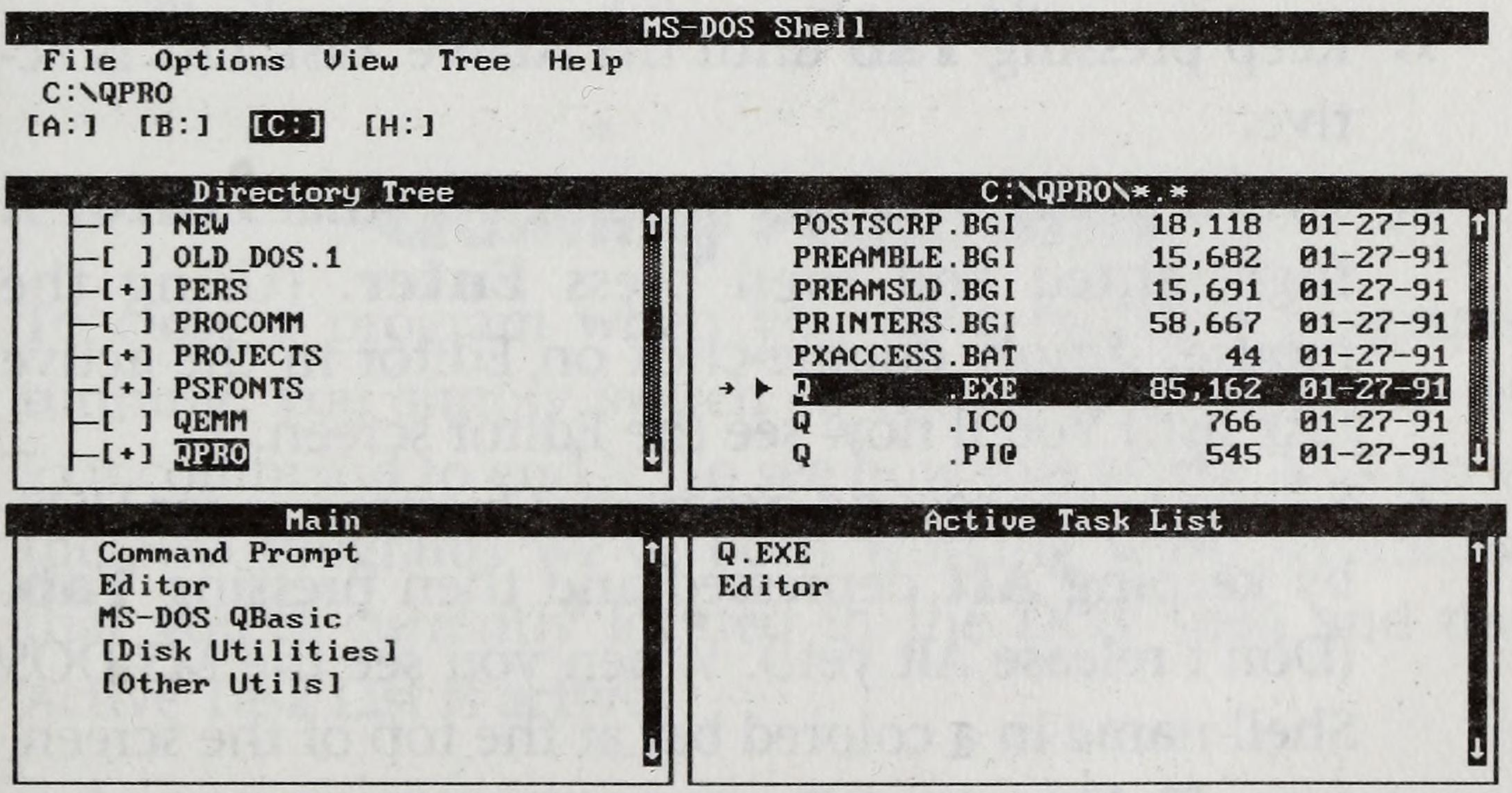


Figure 16.1 The Active Task List area

At this point, the two programs—MS-DOS Editor and Quattro Pro for DOS—are running. In the next section, we'll show you how to switch back and forth between them.

Tip

You can add a program to the Active Task List without running it. To do this, select the appropriate executable file, such as Q.EXE, and press Shift+Enter. If you are using a mouse, hold down Shift and double-click on the executable file.

Switching Between Programs

Now that you have two programs running, we'll show you how to switch between them. To follow along, start from the DOS Shell screen shown in Figure 16.1 and perform the following steps:

1. Keep pressing **Tab** until the Active Task List is active.
2. Switch to the Editor by pressing ↓ so that **Editor** is highlighted and then press **Enter**. (Using the mouse, simply double-click on Editor in the active task list.) You'll now see the Editor screen.
2. Switch to the second program, Quattro Pro for DOS, by keeping **Alt** depressed and then pressing **Tab**. (Don't release Alt yet!). When you see the MS-DOS Shell name in a colored bar at the top of the screen, press **Tab** again. When you see Q.EXE in the colored bar, release the **Alt** key. DOS activates Quattro Pro and its screen now appears.

3. Continue to switch between programs—QPro, Editor, and Dos Shell—by keeping **Alt** depressed while pressing **Tab**. When you see Editor in a colored bar at the top of your screen, release the **Alt** key. The Editor screen now appears. Press **Alt+Tab** again, release the **Alt** key, and the Quattro Pro screen reappears.
4. Press **Ctrl+Esc** to return to the DOS Shell.

This example demonstrates that:

- You can activate any program in the DOS Shell Active Task List by highlighting it and pressing Enter or by double-clicking on it with your mouse.
- You can switch between active programs by holding down Alt and pressing Tab until the name of the program you want is displayed. To activate the program, release Alt.
- You can return to the DOS Shell screen at any time by typing Ctrl+Esc. Alternatively, you can hold down Alt and press Tab until the name MS-DOS Shell appears, and then release Alt.

Quitting Programs

To close a program when you have multiple programs running, you simply switch to the program and use its Exit command to end it. To see how this works, let's close the two programs we've been working with. Assuming that you're currently located in the DOS Shell and the Active Task List is active:

1. Move to the Editor by keeping **Alt** depressed and pressing **Tab** until you see **Editor** in the colored bar at the top of the screen.

2. Close the Editor by selecting the **File Exit** command from the Editor menu. (If you've been working in the Editor, you'll be asked to save the current file.) You'll be returned to the DOS Shell, where Editor no longer appears in the Active Task List.

3. Close Quattro Pro for DOS by keeping **Alt** depressed and pressing **Tab** until Q.EXE appears in the colored bar at the top of the screen. Select the **File Exit** command for the Quattro Pro menu, and you'll see:

```
Press any key to return to MS-DOS Shell....
```

4. Press any key such as **Esc** and you'll be returned to the DOS Shell where Q.EXE has been removed from the Active Task List.

Figure 16.2 The DOS Shell won't exit when a program is running.

Tip

If a program locks up when you are running multiple programs, use Alt+Tab or Ctrl+Esc to return to the DOS Shell. Then select the program name in the Active Task List and press Del. This ends the program.

Before you can leave the DOS Shell, you must close all open programs. If you attempt to quit the DOS Shell without first quitting all programs, the error message shown in Figure 16.2 is displayed. Press Enter to clear this error message, quit all programs, and then select File Exit (or Alt+F4) to leave the DOS Shell.

◆ *Lesson* ◆

17

Working at the DOS Command Line

The Purpose of the DOS Shell

Until now, you've learned how to manage the power of DOS by using the DOS Shell. In truth, though, the DOS Shell is something of a facade. When you look behind the scenes, you'll discover that the primary purpose of the DOS Shell is to run the programs located in your C:\DOS directory. Each command you enter or operation you perform in the DOS Shell corresponds to one of these programs.

In fact, you don't need the DOS Shell to run the programs located in your C:\DOS directory. You can run them directly from the DOS command line. Indeed, you'll probably find that the more you use DOS, the more tempted you'll be to circumvent the DOS Shell and work directory from the command line. Most veteran users find this method of interacting with DOS to be easier and more efficient; they also find that their commands tend to execute more quickly.

Understanding COMMAND.COM

The commands you'll use most often at the DOS command line are contained within (internal to) COMMAND.COM. This file is loaded into memory automatically each time you boot your computer. COMMAND.COM includes most of the file and directory management commands that are the "nuts and bolts" of the DOS command language. They include commands such as Dir, Copy, Rename, and Del, which are discussed in more detail in the sections that follow.

The remaining commands in the DOS command language—including Format, Edit, Diskcopy, and others—are external to COMMAND.COM, and they are stored in the .EXE and .COM files in your C:\DOS directory. We'll also touch upon many of the more commonly used of these in the sections that follow.

Using Commands

Entering a Command

Remember from Lesson 1 that the first thing you see when your computer has finished booting is the DOS prompt. On most computers, it looks like this:

```
C:\>
```

Tip

You can access the DOS prompt from within the DOS Shell by choosing "Command Prompt" from the program-list area or by pressing Shift+F9. To return to the DOS Shell at any time, type exit and press Enter.

To see how to enter a command, let's try entering the Dir command, which displays a list of the subdirectories and files in the current directory.

1. At the command line, type **dir**. What you're doing here is entering the command name followed by any parameters. (Parameters are discussed in the next section.)
2. Press **Enter** to execute the command. DOS displays a listing similar to the following:

```
 Volume in drive C is DOS_6
 Volume Serial Number is 16E9-2625
 Directory of C:\

 123             <DIR>          12-18-92  12:47p
 DOS             <DIR>          12-18-92  12:50p
 QPRO            <DIR>          12-18-92  12:55p
 QUICKEN         <DIR>          12-18-92  12:55p
 WP51            <DIR>          12-18-92  12:59p
 OLD_DOS  1      <DIR>          04-06-93  10:06a
 COMMAND  COM        52925 03-10-93   6:00a
 AUTOEXEC BAT          297 03-10-93   6:00a
 MOUSE    SYS        55160 09-14-92  12:02p
 CONFIG   SYS          571 04-15-93   9:56a
        11 file(s)     142689 bytes
                    389955584 bytes free
```

3. To clear the dir information on your screen, use the Cls command by typing **cls** and then pressing **Enter**. DOS clears your screen leaving only the DOS prompt C:\>.

COMMAND SYNTAX

Most DOS commands have three parts: a *command name*, *parameters*, and *switches*. In some cases, the parameters and switches are optional. For example, building on the

Dir command example in the previous section, let's suppose you want to get a list of .EXE files in the current directory. But, instead of the single-column list shown in the previous section, you want to see a multiple-column list so that more filenames can appear on your screen. To do this, use the following command:

```
dir *.exe /w
```

The command name is again **dir**, and the parameter for the command is ***.exe**, which is an optional parameter that uses a wildcard pattern to describe only the .EXE files in the current directory. Finally, the last argument for the command is the optional **/w** switch. Notice that a switch begins with a forward slash (/) followed by a letter. This particular switch causes a multiple-column listing to be displayed for the Dir command. Here is the output from this command:

```
 Volume in drive C is DOS_6
 Volume Serial Number is 16E9-2625
 Directory of C:\DOS
NLSFUNC.EXE    ATTRIB.EXE     CHKDSK.EXE     EXPAND.EXE     MSD.EXE
QBASIC.EXE     EDLIN.EXE      RESTORE.EXE    DEBUG.EXE      FDISK.EXE
DEFRAG.EXE     DOSSWAP.EXE    RECOVER.EXE    DOSSHELL.EXE   MEM.EXE
XCOPY.EXE      DELTREE.EXE    MOVE.EXE       BACKUP.EXE     SMARTDRV.EXE
COMP.EXE       FASTHELP.EXE   FASTOPEN.EXE   POWER.EXE      PRINT.EXE
JOIN.EXE       EXE2BIN.EXE    SHARE.EXE      SETVER.EXE     APPEND.EXE
DELOLDOS.EXE   FC.EXE         FIND.EXE       LABEL.EXE      SMARTMON.EXE
SORT.EXE       MWBACKUP.EXE   REPLACE.EXE    SUBST.EXE      DBLSPACE.EXE
INTERLNK.EXE   INTERSVR.EXE   MSCDEX.EXE     EMM386.EXE     MEMMAKER.EXE
SIZER.EXE      UNDELETE.EXE   MWUNDEL.EXE    MWAV.EXE       MSBACKUP.EXE
MWAVTSR.EXE    MSAV.EXE
        52 file(s)     2638465 bytes
                     389955584 bytes free
```

Pausing a Command

You can pause the execution of a command by pressing Ctrl+S or the Pause key on your keyboard. The command

will stop executing until you press any key (except Pause). Let's say you just used the Dir command to list a large directory. But when you pressed Enter to activate this command, the names of files and directories scrolled by so quickly you couldn't read them. To pause the command enabling you to read the contents of the list, press the **Pause** key. DOS will pause the listing until you press any key other than Pause.

Cancelling a Command

If you change your mind about a command you've just invoked, you can cancel the action at any time by pressing Ctrl+c or Ctrl+Break. Even if the command has already started executing, pressing one of these key sequences suspends further execution. Again, let's assume you just used Dir to get that large directory listing, but then realized you didn't need it. Rather than waiting for the entire listing to be displayed, press **Ctrl+c** in midstream. Further execution of the command will cease and you will be returned to the DOS prompt.

Repeating the Last Command

DOS supports two special keys that let you easily repeat the last command you've entered at the DOS prompt. These are:

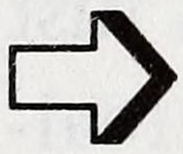
- F1, which enters your last command one letter at a time. (You can achieve the same effect by pressing →).
- F3, which enters all of the characters in the previous command.

You might want to try the following:

1. Type **dir** and press **Enter** to see a listing of the current directory.
2. Press **F3** and DOS displays the last command you typed, dir.
3. To see a listing of your DOS directory, edit the command by pressing the **Spacebar** and typing **C:\dos*.***. At this point your command line should look like

```
dir C:\dos\*.*.
```

4. Press **Enter** to execute this command and DOS displays a list of all files and subdirectories in the C:\DOS directory.



Note: If Doskey is active, the keys F1, →, and F3 will not work.

Using Doskey

Doskey is an innovation first introduced in DOS 5.0 that makes the command line much easier to use. Doskey maintains a list of your most recent commands, which you can access quickly to retrieve and reuse an old command. To see how Doskey works, try the following:

1. At the command line, type **doskey** and press **Enter**. You'll see the message "Doskey Installed." Doskey will now be in effect until you shut off your computer.

Tip

By adding the word doskey to your AUTOEXEC.BAT file, Doskey will run automatically when you start your PC. Lesson 30 explains how to modify your AUTOEXEC.BAT.

2. Type **cd \dos** and press **Enter** to move to the DOS directory. Type **dir** and press **Enter** to see the contents of the DOS directory. Then type **cd** and press **Enter** to move to the root directory C:\.
3. Use Doskey to view previous commands. Press ↑ and Doskey displays the last command you typed, cd\. Press ↑ again and you'll see dir; press ↑ a third time and you'll see cd \dos. Press ↓ so that dir appears on the command line.
4. Modify the command so that it appears as dir /p by pressing the **Spacebar** and then typing **/p**. Press **Enter** and DOS displays the contents of the current directory (the root directory C:\) one screen at a time. Keep pressing any key such as **Esc** until you've seen the entire directory listing.

Tip

Press F7 and Doskey will display a numbered listing of your most recent commands.

Getting Help at the Command Line

Suppose you're having difficulty with a command, such as the Dir command, or you simply want to learn a little

more about it. Here's how to get help at the command line:

1. Get help for a specific command by including the command name when you run the Help command. In this instance, type **help dir.**
2. Press **Enter** and you'll directly access DOS's help for the Dir command, including a brief description of what the command does, its syntax, and any switches it may support.
3. To leave help, select the **File Exit** command from the Help menu.

◆ *Lesson* ◆

18

Working with Drives and Directories

Changing to a Different Drive

You can refer to a specific drive at the DOS command line by using its drive letter followed by a colon (:). For example, try the following:

1. Insert a formatted floppy disk in drive A.
2. Type **a:** and press **Enter**. DOS changes from the current drive C and makes A the current drive.
3. Switch back to drive C by typing **c:** and pressing **Enter**.

When you change to a drive for the first time, DOS shows the root directory on that drive as the current directory. If, on the other hand, you have accessed other directories on that same drive during the current session, DOS returns to the directory you accessed most recently on the drive.

Changing to a Directory

In Lesson 2, you learned how to change from one directory another by using the DOS Shell. You can do the same thing at the DOS command line by using the Chdir command (cd for short): Type **cd** followed by a space and the path to the directory.

As you may recall from Lesson 2, the path to a directory is represented by a list of the directory names that form the connection between the root directory (\) and the directory in question. The directory names in the path are separated by backslashes (\).

If your root directory C:\ is the current directory on your PC and you want to change to DOS as your current directory, use the following command:

```
cd dos
```

Taking this example a step further, let's suppose that your DOS directory has a subdirectory named FILES. To change to this directory from your root directory, use this command:

```
cd dos\files
```

To move one level up the directory tree to the parent directory, use the cd command with .. (dot-dot). Building on the previous example, if your C:\DOS\FILES directory is current, you can use the following command to move up one level to C:\DOS:

```
cd ..
```

You can move to *any* directory on a disk, regardless of which directory is current, by following the cd command

with a backslash (\), indicating the root directory, and the path of the directory. So, if the current directory is C:\DOS\FILES and you want to move to another directory named C:\WP51\DOCS, use:

```
cd \wp51\docs
```

Regardless of which directory or subdirectory you're currently located in, you can move back to the root directory of the current drive such as C:\ using:

```
cd\
```

Viewing the Contents of a Directory

As you've already seen, if you've followed the exercises in Lesson 17, the Dir command generates a list of the files and subdirectories in a directory. Lesson 19 also provides more examples of how to use the Dir command.

Creating Directories and Subdirectories

If you place all your files in a single directory, it will soon become so crowded that finding your files will become a nightmare. To prevent this, plan a workable structure of parent directories, create them, and then create any needed subdirectories beneath those parent directories. With a logical structure of directories in place, you shouldn't have any trouble locating important files.

To create a directory at the command line, you use the Mkdir command (md for short). This command creates a

subdirectory of the current directory using the name you specify. For example, let's create a directory of your root (\) called MYFILES with two subdirectories called BUSINESS and PERSONAL using these steps:

1. Type **cd** and press **Enter** to make the root directory current. Your command line should read C:\>.
2. Create the directory by typing **md myfiles** and pressing **Enter**.
3. Move to the myfiles directory by typing **cd myfiles** and pressing **Enter**. Your command line should look like C:\MYFILES.
4. Create the subdirectories. First type **md personal** and press **Enter**. Then type **md business** and type **Enter**.
5. View the subdirectories by typing **dir** and then pressing **Enter**. You should see a directory listing of MYFILES with two subdirectories, BUSINESS and PERSONAL.

Once these directories and subdirectories have been created, you can begin using them immediately. You can start, for instance, using the DOS Copy command (see the next lesson) to copy your personal files to C:\MYFILES\PERSONAL and your business files to C:\MYFILES\BUSINESS.

Viewing a Directory Tree

You can use the Tree command to see a simple diagram of subdirectories within a directory. For example, try this:

1. Type **cd \myfiles** to make sure that myfiles is the current directory.

2. Type **Tree** and press **Enter**. You'll see a listing like this:

```
|___BUSINESS
|___PERSONAL
```

Tip

For the current directory, using **dir*.** generates a list of subdirectories.

Removing Directories

If you make a mistake while creating a directory, or you decide a directory is no longer needed, you can remove it. In DOS 5.0, you can use the Rmdir command (rd for short) from the command line to remove an *empty* directory containing no subdirectories or files. Besides the Rd command, in DOS 6.0 you can also use the new Deltree command to remove a directory and all files and subdirectories within it.

Using the Rmdir command

To see how the Rmdir (rd) command works, try deleting the BUSINESS subdirectory and the MYFILES directory:

1. To delete the BUSINESS subdirectory, type **cd \myfiles** and press **Enter** to make sure you're located in the MYFILES directory, which contains the BUSINESS subdirectory. The command line should read C:\MYFILES. Type **rd business** and press **Enter**. Type **dir** and press **Enter** to confirm that the BUSINESS subdirectory no longer exists.

2. Try deleting the MYFILES directory by typing **cd** and pressing **Enter** to return to the root directory C:\. Then type **rd myfiles** and press **Enter**. The error message

```
Invalid path, not directory, a directory not empty
```

indicates that MYFILES is not empty—it still contains the PERSONAL subdirectory.

To delete the MYFILES directory, you must first use the Del command to delete all files in PERSONAL, and then the Rd command to delete the PERSONAL directory. In DOS 6, another option is to use the Deltree command instead.

Using the DelTree command

To see how the DOS 6 Deltree command works, try deleting the MYFILES directory which still contains the PERSONAL directory:

1. Type **cd ** and press **Enter** to make sure that you're located at the root directory C:\ which contains the MYFILES directory.
2. Type **deltree myfiles** and press **Enter**. You'll see the message

```
Delete directory "myfiles" and all its
subdirectories? [yn]
```

3. Type **y** and press **Enter** to confirm. (Type n to cancel.)

4. View a directory listing of the root directory one screen at a time by typing **dir /p** and pressing **Enter**. Then keep pressing any key such as **Esc** until you've seen the entire listing. Notice that the MYFILES directory no longer exists nor does its PERSONAL subdirectory.

◆ *Lesson* ◆

19

Getting Information about Files

Getting Information about a Specific File

You can use the Dir command to check for the existence of a specific file in a directory. For example, try this approach to check for the existence of README.TXT in your C:\DOS directory:

1. At the command line, type **cd \dos** from any directory. Press **Enter** and DOS moves you to the DOS directory.
2. Type **dir readme.text** and press **Enter**. DOS displays the following listing:

```
Volume in drive C is DOS_6
Volume Serial Number is 16E9-2625
Directory of C:\DOS

README   TXT      61857 03-10-93   6:00a
        1 file(s)      61857 bytes
                  389521408 bytes free
```

Getting Information about a Group of Files

By using the Dir command with a wildcard pattern and/or file extensions you can obtain information about a group of files that have something in common—for example, similar names or matching file extensions. For instance, try accessing a list of the files in your C:\DOS directory that begin with the letter C and have an .EXE extension:

1. At the command line, type **cd \dos** and press **Enter** to make sure you're located in the DOS directory.
2. Type **dir c*.exe** and press **Enter**. You'll see the following listing:

```
Volume in drive C is DOS_6
Volume Serial Number is 16E9-2625
Directory of C:\DOS

CHKDSK   EXE     12907 03-10-93    6:00a
COMP     EXE     14282 04-09-91    5:00a
        2 file(s)      27189 bytes
                  389529600 bytes free
```

In this case, the file extension limits the listing to .EXE files, while C* specifies files only beginning with C regardless of the length of the file name.

Note: If you used the ? wildcard in C??.EXE, you would limit the list to files beginning with C with exactly three characters.

Controlling the Files List

The Dir command also accepts several switches that let you control the structure and scrolling method for the files list generated by the command. Two of the most commonly used switches are:

/p Displays a listing one screen at a time. Press any key to see the next screen.

/w Displays a 5-column listing, allowing you to see all the file or directory names at once.

You saw an example of how to use the /p switch in the previous lesson. To see how the /w switch works, try this:

1. At the command line, type **cd \dos** and press **Enter** to make sure you're located in the DOS directory.
2. Type **dir *.com /w** and press **Enter**. (From any directory, you could use dir c:\dos*.com \w.) You'll see the following listing:

```
Volume in drive C is DOS_6
Volume Serial Number is 16E9-2625
Directory of C:\DOS

FORMAT.COM     KEYB.COM       EDIT.COM     MORE.COM       SYS.COM
UNFORMAT.COM   MIRROR.COM     CHOICE.COM   MODE.COM       MSHERC.COM
DOSSHELL.COM   HELP.COM       ASSIGN.COM   GRAFTABL.COM   DISKCOMP.COM
DISKCOPY.COM   GRAPHICS.COM   LOADFIX.COM  TREE.COM       DOSKEY.COM
MOUSE.COM      COMMAND.COM    VSAFE.COM
      23 file(s)     363805 bytes
                389554176 bytes free
```

In this case, *.COM limits the listing to all .COM files while the /w switch places the files in 5 columns or less.

Tip

At the command prompt, type **help dir** and press **Enter** to see other switches and shortcuts you can use with the Dir command. For instance, you can create listings by size or date of file, display or omit only hidden files, system files, directories, etc.

Viewing and Changing File Attributes

Although you can view and change the attributes of a file through the DOS Shell (see Lesson 12), you can also do so at the command line using the Attrib command. As you may recall from Lesson 12, each file can have one or more of four attributes: A (archive), R (read only), H (hidden), and S (system).

Viewing File Attributes

When you type **attrib** and press **Enter** at your root directory C:\, you'll see a listing similar to the following:

```
        SHR   C:\IO.SYS
        SHR   C:\MSDOS.SYS
  A           C:\COMMAND.COM
  A           C:\MOUSE.SYS
      S       C:\PCTRACKR.DEL
```

Notice that some files have more than one attribute assigned to them. For example, IO.SYS is a hidden, system, read only file.

Changing File Attributes

You can also use the Attrib command to change an attribute for a file. The convention is **attrib** and **+**, to add, or **–** (minus), to delete, the attribute symbol filename.

For instance, let's say you want to protect the README.TXT in the C:\DOS directory from being deleted or changed by assigning it a read only attribute. Here's how to do this:

1. At the command line, type **cd \dos** and press **Enter** to go to the DOS directory.
2. Type **attrib +r readme.txt** and press **Enter**.
3. View the attributes of the files in the DOS directory by typing **attrib** and pressing **Enter**. README.TXT should now have an R before it denoting read only.
4. Turn off the read only attribute by typing **attrib -r readme.txt** and pressing **Enter**.

You can also use the Attrib command on a group of files by using a wildcard pattern in place of a filename. To assign read only status to all text files in your DOS directory, for instance, you would type **attrib +r *.txt**.

◆ *Lesson* ◆

20

Working with Files from the Command Line

Renaming a File

The Rename command (Ren for short) allows you to rename a file at the command line. (In the DOS Shell, you can rename a file using the File Rename command.) As an exercise, let's try changing the name of README.TXT in your C:\DOS directory to DOSREAD.ME:

1. At the command line, type **cd \dos** and press **Enter** to make sure that you're located in the DOS directory.
2. Type **ren readme.txt dosread.me** and press **Enter**. Notice that you use ren then the name of the file to be changed followed by the new name.
3. Type **dir *.me** and you'll see a listing of all files in the DOS directory with a *.ME extension, the only one being DOSREAD.ME.
4. Change the file name back by typing **ren dosread.me readme.txt** and then pressing **Enter**.

The Ren command has a built-in safeguard; it won't let you overwrite a file in the current directory with the same

name. So, if the new name you chose for a file matches that of an existing file in the same directory, you'll see the message:

```
Duplicate filename or file not found
```

You can also use the Ren command in conjunction with DOS wildcard symbols to rename groups of files. For example, follow these steps to rename all .TXT files in your DOS directory with a .PRN extension:

1. From any directory, type **ren c:\dos*.txt *.prn** and press **Enter**.
2. Type **dir\dos*.prn** to see a listing of all files in the DOS directory with a *.PRN extension.
3. Change the file extension back by typing **ren c:\dos*.prn *.txt** and then pressing **Enter**.

Copying Files from the Command Line

At the command line, you can use the Copy command to copy one or more files from one directory to another or to a different drive. The Copy command requires two parameters. The *Source* is the path and name of the file(s) you want to copy; the *Destination* is the path and name (optional) to which you want to copy the file(s).

Tip

Because the DOS Shell is so visually oriented and supports drag-and-drop, many times it's easier to copy files there instead of at the command line. See Lesson 3.

Copying a Single File

To demonstrate how to copy a single file using the Copy command, we'll copy README.TXT from your C:\DOS directory to the root directory (C:\). It's a simple process:

1. At the command line, type **cd \dos** and press **Enter** to make sure you're in the DOS directory.
2. Type **copy readme.txt c:** and press **Enter**. (From any directory, you could type copy c:\dos\readme.txt c:\.) After DOS copies the file, you'll see the message "1 file(s) copied."
3. View the copied file by typing **dir c:*.txt** and pressing **Enter**. The copied file README.TXT will appear in the directory listing of C:\.

When you use the Copy command to copy a file from one directory to another, DOS overwrites any file in the Destination directory that has the same name as the file you are copying. If you know there is a duplicate file in the Destination directory, you can prevent it from being overwritten by changing the name of the source file as it is being copied.

Changing the Name of a File While Copying

The Copy command allows you to change the name of the file being copied in the destination by including a filename as part of the Destination. For instance, try this to copy README.TXT from C:\DOS to the root directory C:\ under the name TEXT.TXT:

1. At the command line, type **cd \dos** and press **Enter** so that you're located in the DOS directory.

2. Type **copy readme.txt c:\text.txt** and press **Enter**. After DOS copies the file, you'll see the message "1 file(s) copied."
3. View the copied file by typing **dir c:*.txt** and pressing **Enter**. The copied file under the new name TEXT.TXT will appear in the directory listing of C:\.

Tip

If you are copying a file from one directory to another on the same drive, you can leave out the drive designator in the Source and Destination parameters for the Copy command. For example, copy \dos\readme.txt \text.txt has the same effect as the command shown above.

Copying a Group of Files

By using a wildcard pattern in the Source parameter of the Copy command, you can copy a group of files that have something in common—matching file extensions, for example. To see how this works, try copying the *.TXT files in your DOS directory to the root directory C:\:

1. At the command line, type **cd \dos** and press **Enter** so that you're located in the DOS directory.
2. View which files you're going to copy by typing **dir *.txt** and pressing **Enter**. If you're working with DOS 6, you'll see a listing of these files:

```
NETWORKS TXT       23444 03-10-93    6:00a
README   TXT       61857 03-10-93    6:00a
APPNOTES TXT        9701 04-09-91    5:00a
OS2      TXT        6358 03-10-93    6:00a
```

3. Copy the files to the root directory by typing **copy *.txt c:** and pressing **Enter**. When DOS finishes copying, you'll see the message "4 file(s) copied."
4. View the copied files in the destination by typing **dir c:*.txt** and pressing **Enter**. The four copied files will appear in the directory listing of C:\. (If you copied README.TXT in the previous exercise, it is written over during this copy operation without confirmation.)

Copying Files to a Different Drive

The Copy command also can be used to copy files from one drive to another. For example, here's how to copy the *.TXT files in your DOS directory to a floppy disk in drive A:

1. Insert a formatted floppy disk in drive A.
2. At the command line, type **cd \dos** and press **Enter** so that you're located in the DOS directory.
3. Copy the .TXT files by typing **copy *.txt a:** and pressing **Enter**. When DOS finishes copying, you'll see the message "4 file(s) copied."
4. View the copied files in the destination by typing **dir a:** and pressing **Enter**.

Deleting Files

Deleting files from the DOS command line is accomplished with the Del command. Simply type **del** followed by a space, and the path and name of the file you want to delete. For example, if you have been following along with the examples in this section, you now have four files (README.TXT, TEXT.TXT, NETWORKS.TXT, and

APPNOTES.TXT) in your root directory that don't belong there. To delete two of these files, make your root directory current (type **cd ** and press **Enter**), then type the following three commands at the DOS prompt, pressing **Enter** after each one:

```
del appnotes.txt
del networks.txt
```

The Del command, too, can be used with a wildcard pattern to delete groups of files. Let's say hypothetically that the four files are the only .TXT files in your root directory. (Check to make absolutely sure these are the only .TXT files in that directory before trying this next example.) To delete these files, type **del *.txt** at the DOS prompt and press **Enter**. At a minimum, APPNOTES.TXT and README.TXT will be deleted.

Moving Files

DOS 6, unlike previous versions, includes a Move command you can use at the command line. The Move command, like Copy, requires two parameters. The *Source* is the path and name of the file(s) you want to move; the *Destination* is the path and name (optional) to which you want to copy the file(s).

Tip

Because the DOS Shell is so visual and supports drag and drop, many times it's actually easier to move files there instead of at the command line. See Lesson 4.

Moving a Single File

To see how to move a single file, try moving README.TXT from the DOS directory to your root directory C:\:

1. At the command line, type **cd \dos** and press **Enter** so that you're located in the DOS directory.
2. Move the README.TXT file by typing **move readme.txt c:** and pressing **Enter**. (From any directory, you could type move c:\dos\readme.txt c:\.) When DOS finishes, you'll see the confirmation message:

```
C:\dos\readme.txt => C:\readme.txt [ok]
```

3. Confirm that README.TXT no longer resides in the DOS directory by typing **dir *.txt** and pressing **Enter** to view all text files in the current directory. View the moved file by typing **dir c:*.txt** and pressing **Enter**. The copied file README.TXT will appear in the .TXT file listing of C:\.
4. Move the README.TXT file back to the DOS directory by typing **move c:\readme.txt c:\dos** and pressing **Enter**. When the move is completed, you'll see the confirmation message:

```
C:\readme.txt => C:\dos\readme.txt [ok]
```

If you move a file to a directory that already includes a file of that name, DOS overwrites the existing file during the move without asking for confirmation. One way to avoid this problem is to change the name of a file while you're moving it.

Changing the Name of a File While Moving

The Move command allows you to change the name of the file being moved by including a filename as part of the Destination. For instance, follow these steps to move README.TXT from C:\DOS to the root directory C:\ under the name NEW.TXT:

1. At the command line, type **cd \dos** and press **Enter** so that you're located in the DOS directory.
2. Type **move readme.txt c:\new.txt** and press **Enter**. After DOS moves and renames the file, you'll see the confirmation message:

```
C:\dos\readme.txt => C:\new.txt [ok]
```

3. Move this file back to your DOS directory under the correct name by typing **move c:\new.txt c:\dos\readme.txt** and pressing **Enter.** After DOS copies the file, you'll see:

```
C:\new.txt => C:\dos\readme.txt [ok]
```

Moving a Group of Files

By using a wildcard pattern in the Source parameter of the Move command, you can move a group of files that have something in common—matching file extensions, for instance, or similar names. Try moving the *.TXT files in your DOS directory to a directory called OLD:

1. Create the OLD directory by pressing **cd** to move to your root directory C:\; type **md old,** and press **Enter.** Then type **cd \dos** and press **Enter** to move to the DOS directory.

2. Move the .TXT files in the DOS directory by typing **move *.txt c:\old** and pressing **Enter**. When DOS finishes moving, you'll see these messages:

```
C:\dos\networks.txt => C:\old\networks.txt [ok]
C:\dos\readme.txt => C:\old\readme.txt [ok]
C:\dos\appnotes.txt => C:\old\appnotes.txt [ok]
C:\dos\os2.txt => C:\old\os2.txt [ok]
```

3. Move these files back to your DOS directory by typing **move c:\old*.txt c:\dos** and pressing **Enter.**

◆ *Lesson* ◆

21

Running Programs from the Command Line

About Executable Files

Perhaps the easiest and most frequent operation you'll perform at the DOS command line is running programs. To run a program, you must know the name of its executable file. As you may recall, executable files have an .EXE or .COM. extension.

For example, try running the MS-DOS Editor from the command line. Since it's executable file EDIT.COM is located in the DOS directory:

1. At the command line, type **cd \dos** and press **Enter** so that you're located in the DOS directory.
2. To run the Editor, type **edit** and press **Enter**. Notice that you don't need to include the file extension .COM. (From any directory, you could use c:\dos\edit to start the Editor.)
3. When you press **Enter**, you'll see the MS-DOS Editor screen. Press **Esc** to clear the dialog box you see. Then leave the Editor by choosing the **File Exit** command from the Editor menu.

Tip

If the path to a directory containing the program you want to run is included in the Path statement in AUTOEXEC.BAT, you can run the program from any directory without including the path before the executable filename. In fact, because the DOS directory is almost always included in the Path statement, you can execute DOS commands from any directory without specifying a path. See Lesson 30 for more on AUTOEXEC.BAT.

Passing Parameters to Programs

You can further control a program when you run it by passing a parameter—such as a file name—when you invoke the executable file. For example, here's how to run the MS-DOS Editor and automatically load README.TXT into it:

1. At the command line, type **cd \dos** and press **Enter** so that you're located in the DOS directory.
2. Type **edit readme.txt**. When you press **Enter**, the MS-DOS Editor starts and the README.TXT file is loaded.
3. Choose the **File Exit** command to leave the Editor.

Making the Leap from the DOS Shell

In various lessons throughout this book, we've shown you how to run different DOS utilities using the program items in the program list area of the DOS Shell. In truth,

however, each of these program items runs an executable program file located in your C:\DOS directory. As you might have guessed, you can run these same programs directly from the DOS command line. Table 21.1 shows the relationship between each program item in the DOS Shell, the program it runs, the DOS command required to run the same program from the command line, and the lesson in which the program is discussed.

To run one the Utilities from the DOS command line, type the appropriate command at the DOS prompt followed by the same parameters you would supply for the command through the DOS Shell dialog box. For instance, typing **format A:** at the command line and pressing **Enter** is equivalent to selecting the Format program item in the DOS Shell and specifying A: as the drive to format in the dialog box that appears. In both cases, DOS runs the FORMAT.COM utility to format the floppy disk in drive A.

Table 21.1 Shell Program Group Options and Corresponding Commands

Program Item	DOS Program	DOS Command	Lesson Number
Editor	EDIT.COM	Edit	13
Disk Copy	DISKCOPY.COM	Diskcopy	15
Backup Fixed Disk	BACKUP.EXE	Backup	25
Restore Fixed Disk	RESTORE.EXE	Restore	26
Quick Format	FORMAT.COM	Format (/q)	14
Format	FORMAT.COM	Format	14
Undelete	UNDELETE.EXE	Undelete	23

Note: There is one exception to the foregoing involving the DOS Shell Quick Format program item. When you select this program item, a dialog box is presented requesting a drive to quick format. The suggested parameter is a: (meaning drive A). When you confirm this dialog box, however, the shell actually runs the FORMAT.COM program and passes the a: parameter followed by the /s switch. To do the same thing at the DOS command line, you need to use the following command: format a: /s.

◆ *Lesson* ◆

22

Command Line Shortcuts

Cutting Down on Typing

Some people object to working from the DOS command line because of the amount of typing involved. Admittedly, typing the path to a directory that is three or four levels deep can be a time-consuming process. There are, however, a few shortcuts that can help to reduce the amount of time you spend typing at the command line.

Current Location Shortcuts

Taking Advantage of the Current Drive

As you know by now, the current drive is the disk drive in which you are presently working. DOS always knows which drive that is and it assumes your commands apply to the directories and files on that drive. Therefore, provided the effect of a command is confined to the current drive, you don't need to include the current drive letter in

the command. For example, the command **copy c:\myfiles\notes.txt c:\work** can become **copy \myfiles\notes.txt \work**.

Taking Advantage of the Current Directory

The current directory, like the current drive, is the one in which you happen to be working, and DOS will always look in this directory automatically when searching for files. Therefore, if the file you are referring to is located in the current directory, you don't have to include the path to the directory in front of the filename. This means that if the current directory is C:\MYFILES\PERSONAL and you want to copy its .TXT files to a disk in drive A, instead of using the command **copy \myfiles\personal*.txt a:**, you can use **copy *.txt a:**.

Tip

DOS substitutes the current directory as the Destination argument for the Copy command, so if you want to copy all the .TXT files in the root directory on drive A to the current directory on drive C, use the command copy a:*.txt.

In addition, if the command you are using affects only files in the current directory, you can leave out the path entirely. For example, let's suppose the current directory is C:\MYFILES\PERSONAL and you want to copy a file named NOTES.TXT to a new file named JULY.TXT in that same directory. Rather than using the long command **copy \myfiles\personal\notes.txt \myfiles**

personal\july.txt, you can simply substitute **copy notes.txt july.txt**.

Subdirectory Shortcut

When referencing a subdirectory of the current directory, it's not necessary to include the full path to the subdirectory in a command. If, for example, the current directory is C:\MYFILES\PERSONAL, and you want to copy all the .TXT files in this directory to a subdirectory named C:\MYFILES\PERSONAL\JULY, use the simple command **copy *.txt july** instead of **copy *.txt \myfiles\personal\july**.

Dot and Dot Dot

Every DOS directory contains a directory called . (dot). This is a special directory that stands for "the current directory." Therefore, when you want to refer to the current directory, you can use a . to do so. Similarly, another directory called .. (dot dot) stands for the parent directory of the current directory. When you refer to this directory, you can use .. for its name. For example, let's suppose the current directory is C:\MYFILES\PERSONAL and you want to copy a file named JULY.TXT in that directory to C:\MYFILES\WORK. You can use the command **copy july.txt ..\work**.

Previous Location Shortcuts

DOS also remembers the last directory that was used on each of your drives. As you know, when you boot your computer, the current directory is the root directory of

each of your disk drives. If you use the Chdir (cd) command to make another directory current, and then switch to another drive, DOS remembers the last directory you were in when you left the old drive. And you can refer to this directory simply by using the drive letter.

For example, let's say that the current directory on drive C is C:\MYFILES\PERSONAL. Then you switch to drive A by typing **a:** and pressing **Enter**. Next suppose you want to copy all the .TXT files in the current directory on drive A to your C:\MYFILES\PERSONAL directory. You can do this with the following command: **copy *.txt c:**.

Notice that you don't have to spell out the \MYFILES\PERSONAL directory name on drive C. DOS remembers the last directory you were in on drive C and responds accordingly.

◆ *Lesson* ◆

23

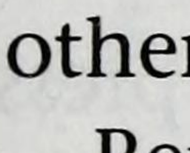
Undeleting Files

Understanding Undelete

Eventually you'll come to know that sinking feeling that comes from deleting a file unintentionally. Fortunately, if you act swiftly with DOS's Undelete utility, you can easily recover a deleted file.

Although Undelete will recover deleted files, it won't recover files that have been overwritten. For example, suppose you inadvertently copy one file on top of another. You won't be able to recover the overwritten file.

Remember too that time is of the essence with Undelete; you should use it as soon as you discover you've deleted a file in error. If you wait, DOS may use the same disk area occupied by the deleted file to store new files and directories.

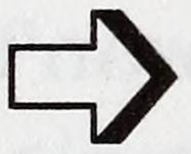


Undelete won't restore a directory that has been deleted. Nor will it restore a file whose directory has been deleted.

When you invoke Undelete, it scans the current directory for files that have been deleted. It then indicates how many deleted files were found and how many it can recover. Finally, it shows you each of the files it found in turn and gives you the option of restoring the file.

Using Undelete in the Current Directory

The best way to see how Undelete works is to delete and then recover a sample file. In this exercise, we'll delete and recover OS2.TXT, which is stored in your DOS directory. (This file contains notes about upgrading from OS/2 to DOS 6 and has no impact on the operation of either operating system.) Make sure you follow the steps carefully. Otherwise, you may lose the file. Perform the following steps:



Note: If you are using DOS 5, substitute APPNOTES.TXT for OS2.TXT in this lesson.

1. From the DOS prompt, type **cd \dos** and press **Enter** to make the DOS directory current.
2. Type **del os2.txt** and press **Enter**. DOS deletes the OS2.TXT file.
3. Type **undelete** and press **Enter**. DOS displays messages like the following:

```
UNDELETE - A delete protection facility
Copyright (C) 1987-1993 Central Point Software, Inc.
All rights reserved.
```

```
Directory:  C:\DOS
File Specifications:  *.*
    Delete Sentry control file not found.

    Deletion-tracking file not found.

    MS-DOS directory contains   1 deleted files.
    Of those,     1 files may be recovered.

Using MS-DOS directory method.

    ?S2     TXT      6358  3-10-93   6:00a ...A
Undelete (Y/N)?
```

If Undelete doesn't start, chances are it wasn't installed on your system when DOS 6 was installed. See "Installing Anti-Virus, Backup, and Undelete After Setup" in Chapter 1 of the *MS-DOS 6 User's Guide*.

4. Type **Y** to recover the file. DOS adds the message:

```
Please type the first character for ?S2     .TXT:
```

5. Type **o**. Undelete adds the letter to the beginning of the filename and displays the message:

```
File successfully undeleted.
```

Note: If the letter you type in step 5 would cause two files to have the same name in the same directory, Undelete will disallow the operation. Instead, it will display a warning message and give you another chance to provide an alternate first letter.

Using Undelete Outside the Current Directory

Note in the previous exercise that you were asked to make the DOS directory current before using Undelete. Although this is a good idea, it isn't entirely necessary; you can use Undelete to recover files located outside the current directory. All you need to do is supply the appropriate path to the directory whose files you want to recover. For example, imagine the root directory (C:\) is current, and you want to recover the deleted files in C:\DOS. To do this, type **undelete c:\dos** and press **Enter**, then respond to the messages as described above.

Selective Use of Undelete

You can also be more selective with Undelete through the use of wildcard characters. For example, suppose you want to recover only those files in your DOS directory that have a .TXT extension. To do this, type **undelete c:\dos*.txt** and press **Enter**, then respond to the messages as before.

◆ *Lesson* ◆

24

Using Higher Levels of Delete Protection

About Delete Tracker and Delete Sentry

In Lesson 23, you learned that the Undelete utility is handy for recovering files that you've inadvertently deleted. But you also learned that it is unable to recover files that have been overwritten. Actually, this drawback only occurs when your PC is configured for the standard level of delete protection (the default). By using a higher level of delete protection—either Delete Sentry or Delete Tracker—you can increase Undelete's ability to recover deleted files. *Delete Sentry* provides the highest level of protection, requiring a small amount of memory and disk space. *Delete Tracker* provides slightly less protection using the same amount of memory but less disk space. The memory required by both is for a portion of the Undelete utility that must be present in RAM when a file is deleted.

If you're using DOS 5, you'll need to use the Mirror utility (MIRROR.COM) to enable a higher level of delete protection. Mirror offers the same level of protection as DOS 6's Delete Tracker.

How Delete Sentry Works

When enabled, Delete Sentry creates a hidden directory named SENTRY on your hard disk. Then, whenever, you delete a file, the memory-resident portion of Undelete moves the file from its current location to the SENTRY directory without changing the file's entry in the File Allocation Table (FAT, a special file that tells DOS where directories and files are stored on the disk). If you later undelete the file, DOS moves the file back to its original location.

The size of the SENTRY directory is limited to approximately 7 percent of your hard disk space. If the files you delete would cause SENTRY to exceed this limit, older files are deleted from the directory until enough room is available for the new ones.

How Delete Tracker Works

With Delete Tracker, a hidden file named PCTRACKR.DEL is created on the root directory of your hard disk. When you delete a file, Undelete records the location of the file in PCTRACKR.DEL. It also updates the FAT telling DOS that the location of the file is now available for another file. When you later attempt to undelete the file, the file's contents will only be recovered if another file hasn't been placed in the same location. Even still, a portion of the overwritten file may be recoverable.

Delete Tracker requires only a small amount of disk space for the PCTRACKR.DEL file. But because Delete Tacker's method of delete protection is inferior to Delete Sentry's, you should use Delete Tracker only if a small amount of disk space is available on your system.

Enabling Delete Sentry Protection

To enable Delete Sentry protection, type **undelete /s** at the DOS prompt and press **Enter**. You then see the following message:

```
UNDELETE - A delete protection facility
Copyright (C) 1987-1993 Central Point Software, Inc.
All rights reserved.

UNDELETE loaded.

Delete Protection Method is Delete Sentry.
Enabled for drives : C
```

With Delete Sentry protection in place, when you attempt to recover deleted files using Undelete (see Lesson 23), you're not prompted for the first letter of a deleted file. Instead, Undelete gets the complete filename from the SENTRY directory.

Tip

Delete Sentry will work most effectively if enabled automatically when you start up your PC. You can do this by including the **undelete /s** command in your AUTOEXEC.BAT file. (DOS reads this file automatically when you start your computer.) See Lesson 30 for more details on entering commands in AUTOEXEC.BAT.

Enabling Delete Tracker Protection

If you have little available disk space on your system, you might prefer to enable Delete Tracker protection rather than Delete Sentry protection. Use the /t switch with Undelete to enable Delete Tracker. For example, type **undelete /tc** and press **Enter** to enable Delete Tracker protection for drive C.

Each time you delete a file, the memory-resident portion of Undelete makes a new entry to PCTRACKR.DEL, causing the file to grow in size. Over time, this file can become quite large, eating up valuable disk space. To control the size of the tracking file, you can limit the number of transactions it contains. To do this, follow the /t parameter with a drive letter, a dash (-), and the number of deletions to record—for example, **mirror /tc-25** limits PCTRACKR.DEL to the 25 most recent deletions. As a new deletion is recorded, the oldest deletion is cast away.

To unload the memory-resident portion of Undelete from memory, run Undelete with the /u parameter—for example, **undelete /u**.

Setting Parameters for Undelete

Besides the /s, /t, and /u parameters already mentioned in this lesson, the Undelete utility supports several additional parameters that control its operation. Here is a list of many of these parameters and what they do:

- **/list** Shows a list of deleted files in the current directory but does not restore them.
- **/all** Recovers all deleted files in the specified directory without prompting you for

confirmation. If the Delete Sentry method of protection is enabled, Undelete will use it. If not, Undelete will use the Delete Tracker method when enabled. If neither method is enabled, all deleted files will be recovered with a pound sign (#) as the first letter in the filename. If, upon recovering a file, Undelete detects another file with the same name exists in the current directory, it substitutes an alternate letter in place of the # sign until a unique filename results. The letters tested and the order of substitution are as follows: %&-0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ.

- **/dos** Recovers only those files in the current directory that DOS shows as having been deleted. If a deletion tracking file is present, it will be ignored.
- **/ds** Recovers only those files listed in the SENTRY directory produced by Delete Sentry.
- **/dt** Recovers only those files listed in the PCTRACKR.DEL file produced by Delete Tracker.
- **/purge** Deletes the contents of the SENTRY directory.

The UNDELETE.INI File

The Undelete utility also reads parameters stored in the UNDELETE.INI file. UNDELETE.INI resides in the C:\DOS

directory. Follow these steps to learn more about the contents of this file and how to edit it:

1. Type **help undelete** at the DOS prompt and press **Enter**.
2. Press **Enter** to select the **Notes** option.
3. Press **PgDn** as necessary to bring the information about UNDELETE.INI into view.
4. Select the **File Exit** command to leave help and return to the DOS prompt.

◆ *Lesson* ◆

25

Backing Up Files

Why Back Up?

It's always a good idea to back up your data periodically. In MS-DOS, "backing up" simply means copying data from one disk to another. Then, if a floppy or hard disk fails, you'll have a backup copy from which you can restore the lost data.

DOS 6 offers a powerful Backup utility (BACKUP.EXE) that makes backing up far easier than ever before. This all-in-one utility lets you back up and restore files at a speed that rivals many third-party programs designed expressly for this purpose.

Backing up files is covered in this lesson. Restoring files is covered in Lesson 26.

This lesson assumes you're using the DOS 6 version of Backup, far different from the DOS 5 version. If you have DOS 5, you'll need to use two utilities for backing up and restoring files: Backup (BACKUP.EXE) and Restore (RESTORE.EXE).

Using Backup

The Backup utility is most frequently used to back up files located on your hard disk to one or more floppy disks in drive A or B. Typically, these files are stored in compressed form on the backup disks, although you can opt to store them in standard MS-DOS format if you like. When you later restore the files to your hard disk, Backup decompresses the backed up files and restores them to their former locations.

Tip

Backup cannot back up to a tape drive. To accomplish this, you'll need to use a program like Symantec's Norton Backup.

In the exercises that follow, we'll show you how to perform a backup of all the files in your DOS directory that begin with the letter *M*. Before you can use Backup, though, you must first configure it for your system.

Configuring Backup

The first time you start Backup, it takes you through a configuration sequence. It starts by testing your system for the various types of hardware—video, mouse, and backup devices (floppy drives). It then performs a compatibility test consisting of a small backup and compare. To carry out the compatibility test, you'll need 2 floppy

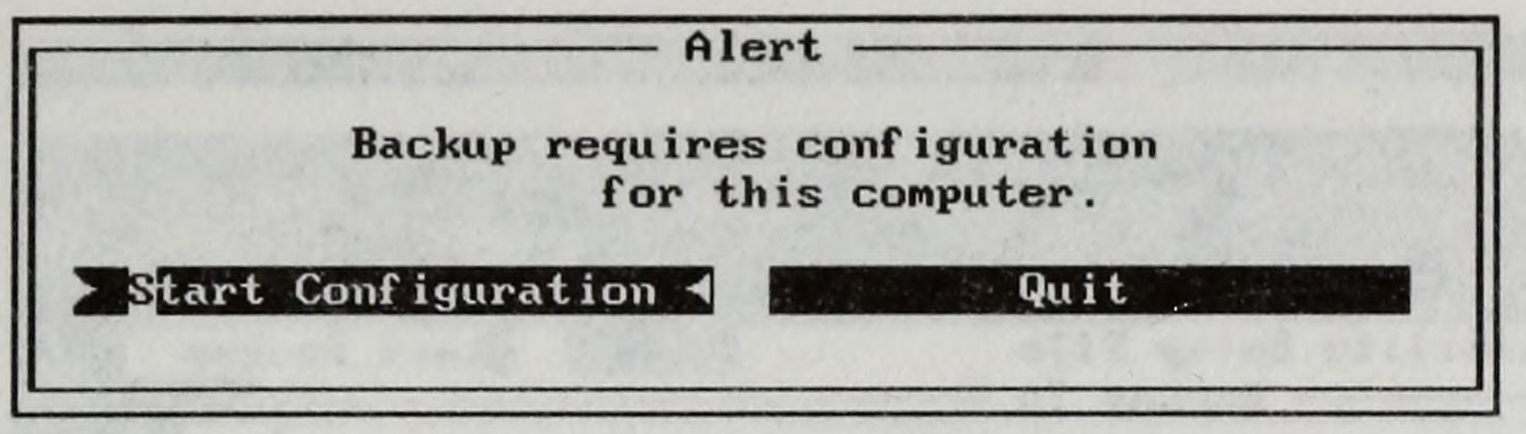


Figure 25.1 The initial configuration message

disks of the same type. Here are the steps to start the sequence:

1. To start Backup, type **msbackup** at the DOS prompt and press **Enter**. Figure 25.1 shows the initial message that appears.

If Backup doesn't start, chances are it wasn't installed on your system when DOS 6 was installed. See "Installing Anti-Virus, Backup, and Undelete After Setup" in Chapter 1 of the *MS-DOS 6 User's Guide*.

2. Press **Enter** to select the **Start Configuration** option. Backup tests your system and shows a series of screens indicating the configuration settings it finds.
3. Press **Enter** at each screen to accept the settings you see and continue the configuration process. Eventually you'll see the screen in Figure 25.2 asking you to specify the floppy drive to use for the compatibility test.
4. If you want to have Backup conduct the test using high-density 5.25-inch floppies in drive A, press

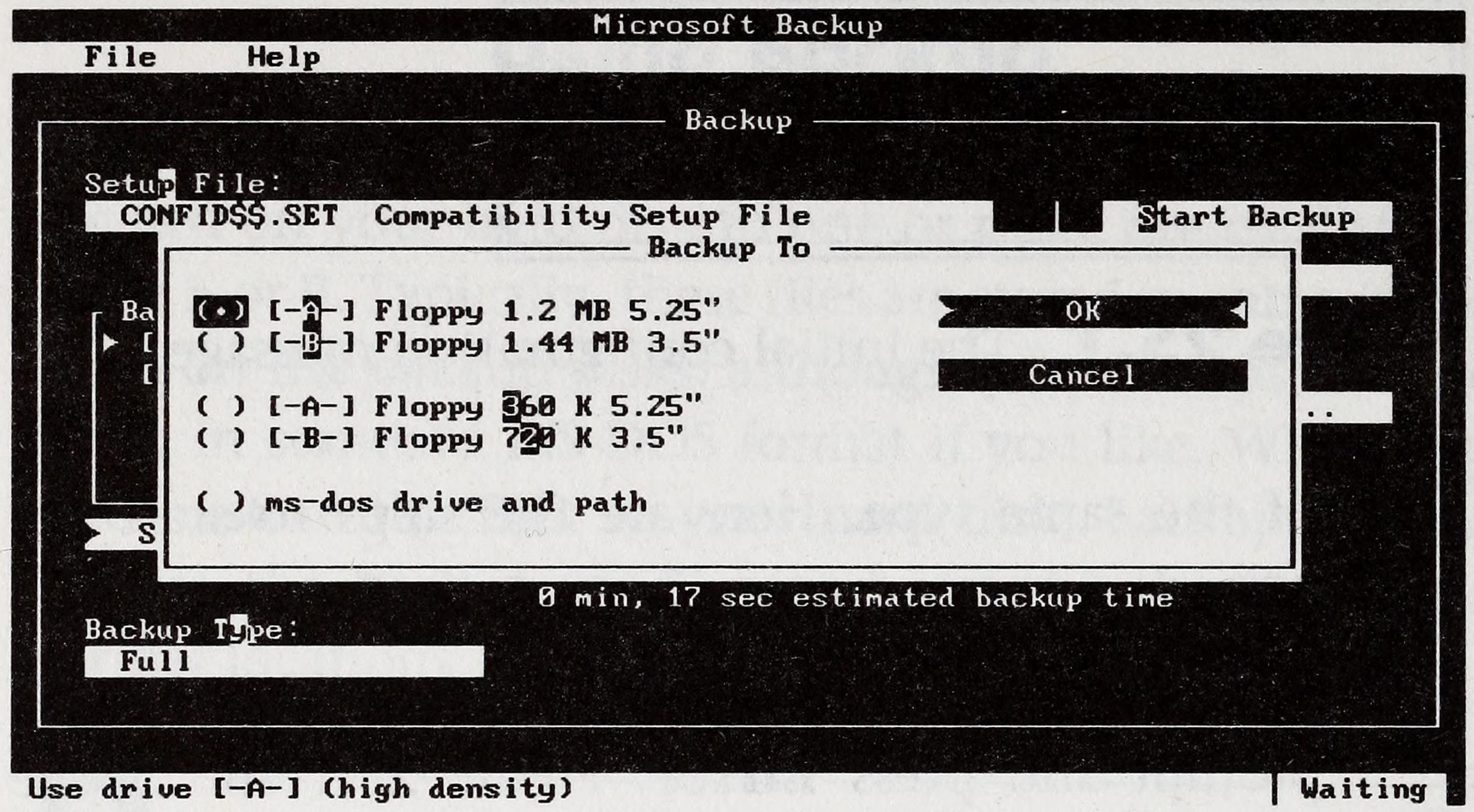


Figure 25.2 Specifying a floppy drive for the compatibility test

Enter. Otherwise, press **Tab** as necessary to move to the appropriate option; press the **Spacebar** to select it, and then press **Enter** to continue. Backup will proceed to carry out the small backup and compare, asking you to insert the two floppy disks at various intervals.

5. Press **Enter** and insert the floppies as necessary to carry out the full test. When the test is complete, you'll see the compatability message in Figure 25.3.
6. Press **Enter** to clear the compatibility message. Backup displays the Configure dialog box.

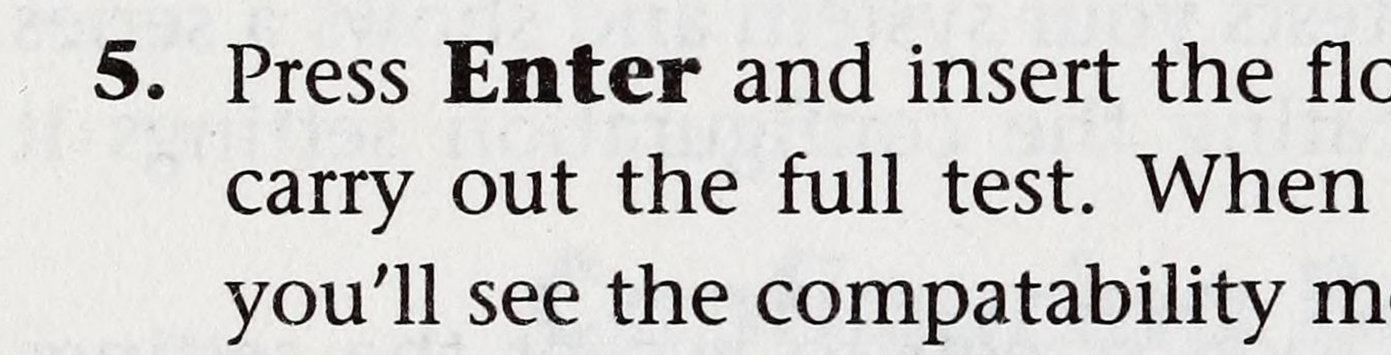


Figure 25.3 Compatibility message

7. Press **Enter** to select the **Save** option. Backup saves the current configuration settings to disk and displays its main screen.

Tip

If you try to run the DOS version of Backup from within a DOS session in Windows, you may receive the message "DMA buffer size too small." If so, place the following line in the [386Enh] section of your SYSTEM.INI file.

```
dmabuffersize=32
```

See the *MS-DOS 6 User's Guide* for more details.

Backing Up Files in the DOS Directory

In this section we'll take you though an exercise that shows you how to back up the files in your DOS directory that begin with the letter *M* to floppy disks in drive A. To follow along, you'll need three blank 5.25-inch, high-density floppy disks. (If you prefer, you can substitute two 3.5-inch high density disks and specify drive B in place of drive A.) Perform the following steps:

Note: This exercise assumes that you've configured Backup as described in the previous exercise.

1. If Backup isn't already running, start it by typing **msbackup** at the DOS prompt and press **Enter**. Figure 25.4 shows Backup's main screen.
2. Press **Enter** to choose the **Backup** option. The Backup dialog box appears showing you all the

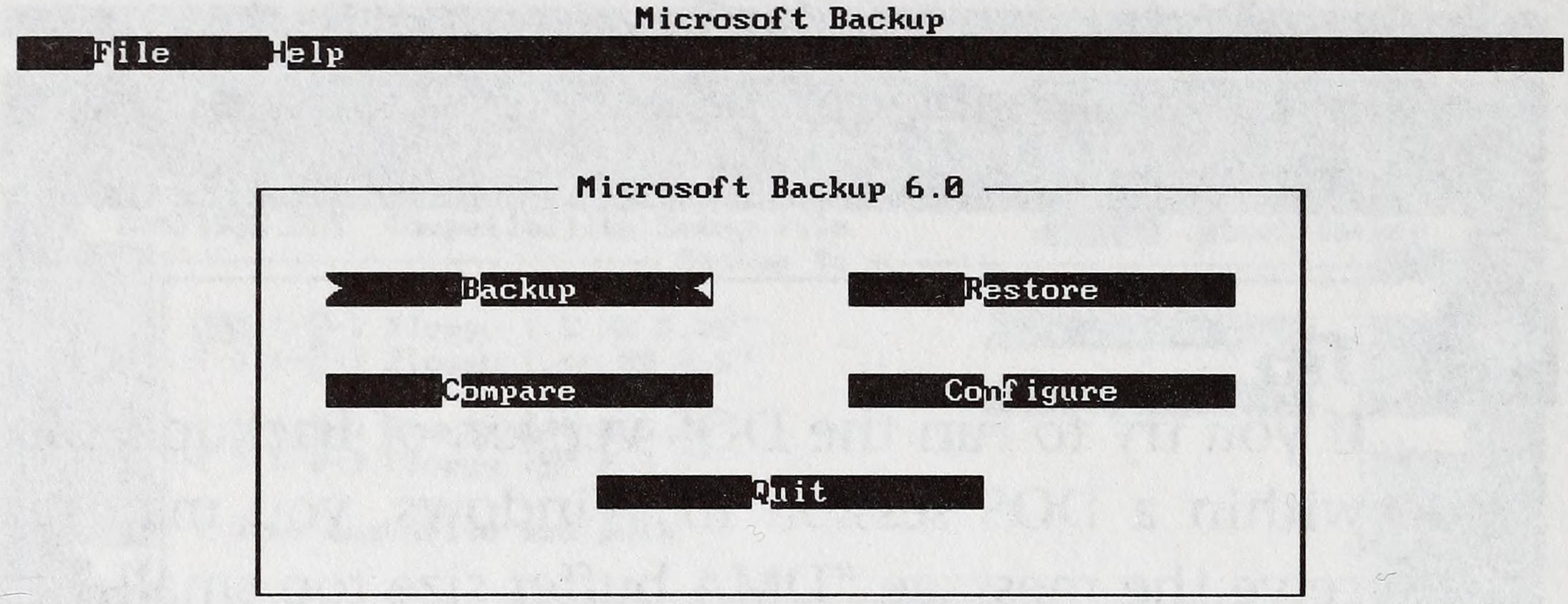


Figure 25.4 Backup's main screen

current settings that will be applied when performing a backup.

3. Press **Tab** and then **Enter** to choose the **Select Files** option. Figure 25.5 shows the screen that appears.

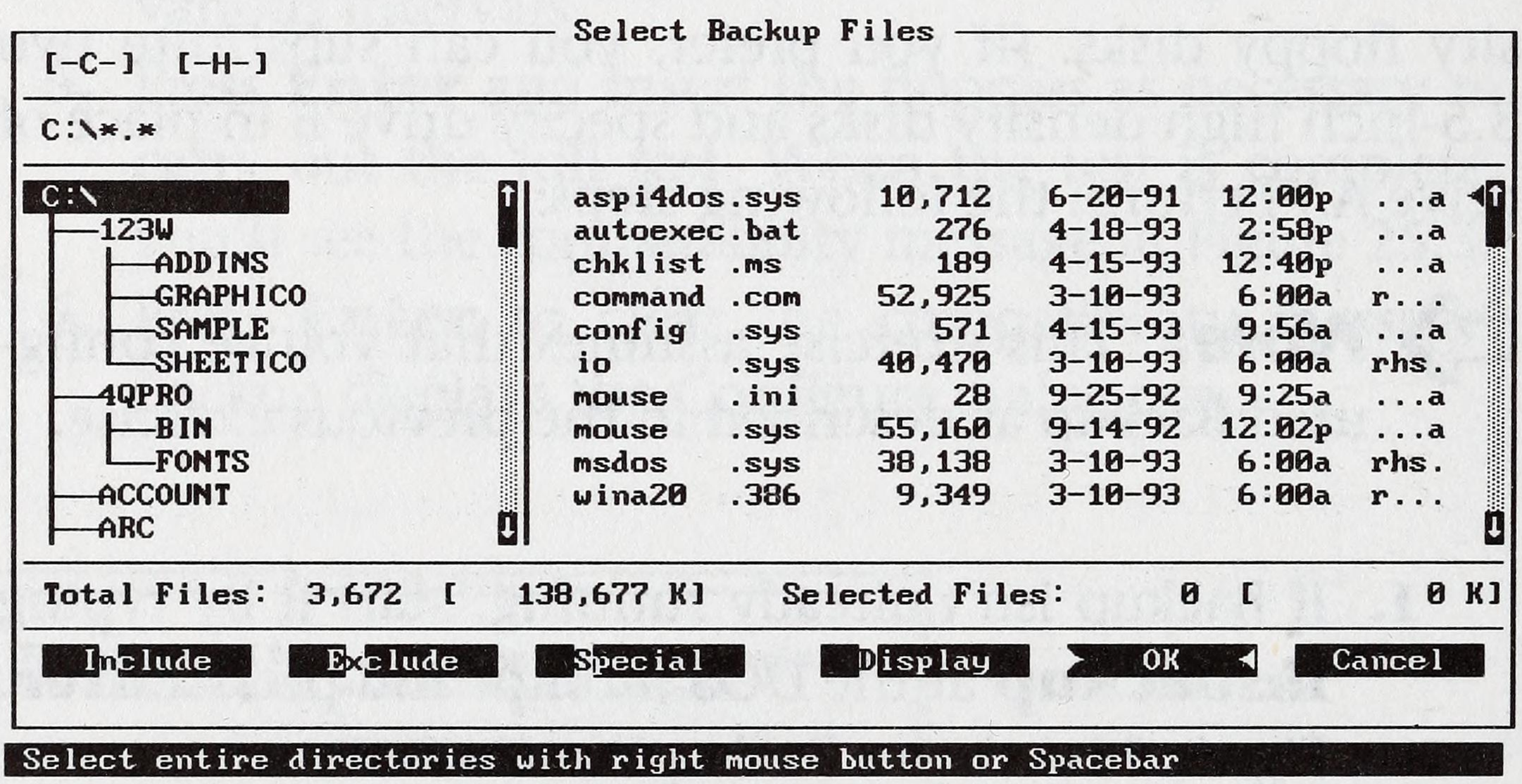


Figure 25.5 Selecting files for backup

4. Press ↓ as necessary to highlight the **DOS** directory in the directory list at the left of the screen.
5. Press **Tab** to move to the file list at the right of screen.
6. Press **m** to highlight the first file in the DOS directory that begins with the letter *M*.
7. Press the **Spacebar** as necessary to select all the files that begin with *M*; a check mark will appear next to a file when it is selected. Backup will move the highlight down the list by one file with each key press. (If you inadvertently select an incorrect file, highlight the file and press the **Spacebar** again to deselect it.)
8. Press **Enter** to choose the OK option and return to the Backup dialog box.
9. If you're backing up to 3.5-inch high-density floppies in drive B, press **Tab** twice to highlight the Backup To option and press **Enter** to select it. (If you're using 5.25-inch floppies for this exercise, skip to step 11.) A screen like the one in Figure 25.2 appears.
10. Press **Tab** to move to the **Floppy 1.44 MB 3.5"** option, press the **Spacebar** to select it, and then press **Enter** to return to the Backup dialog box.
11. Press **Tab** as necessary to highlight the **Start Backup** option and press **Enter** to select it.
12. Insert the floppies when prompted and press **Enter** as necessary to proceed through the backup. (If the floppies you're using already contain data, you'll need to choose the **Overwrite** option when alerted of that fact.) Be sure to label the disks in the order you use them as you go. When the

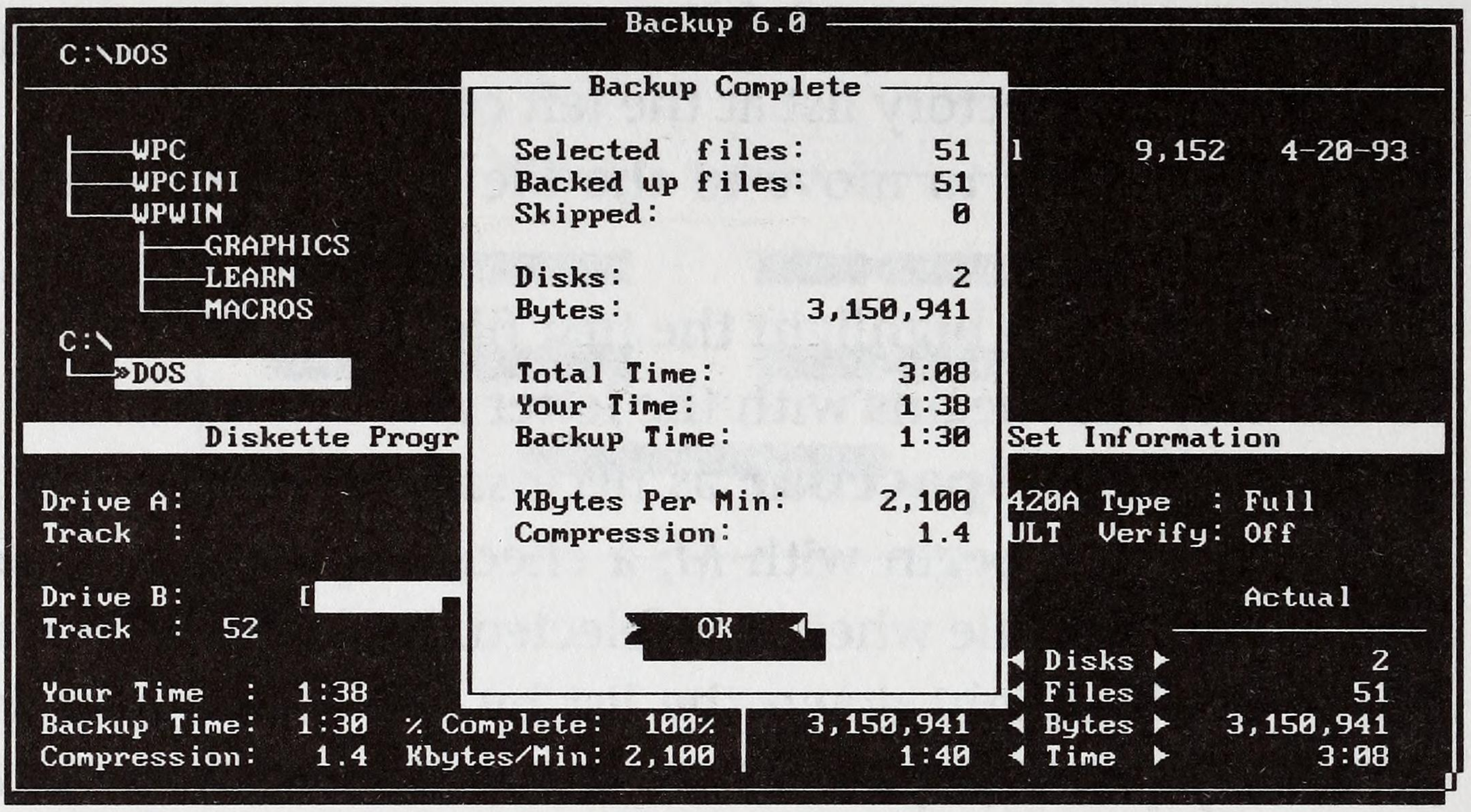


Figure 25.6 Backup Complete message box

backup is complete, you'll see a message box like the one in Figure 25.6.

13. Press **Enter** to clear the message box and return to Backup's main screen.

14. If you wish to exit Backup at this time and return to the DOS prompt, press **Tab** until the **Quit** option is highlighted and then press **Enter**.

◆ *Lesson* ◆

26

Restoring Files

Understanding Restore

As explained in Lesson 25, the Backup utility lets you restore files that you've previously backed up. Typically, you'll want to restore files to their original location on your hard disk. But you can also restore them to another location as well—for example, to another directory or another drive.

Restoring Files to Their Original Locations

In the previous lesson, you learned how to use the Backup utility to back up the files in your DOS directory that begin with the letter *M*. In this lesson, you'll learn how to restore them. (Obviously, these files don't need restoring, but it will serve as a good example.)

Recall that in the previous lesson, it was suggested that you label each backup disk as to the order in which it was used. Now, as you prepare to restore files, Backup will prompt you to insert these disks in the same order. With

the backup disks you created in the previous lesson handy, perform the following steps:

1. If Backup isn't already running, start it by typing **msbackup** at the DOS prompt and press **Enter**. Figure 25.4 (in the previous lesson) shows Backup's main screen.
2. Press **Tab** until the **Restore** option is highlighted and then press **Enter** to select it. Figure 26.1 shows the dialog box that appears.
3. Press **Tab** twice to highlight drive **C** in the Restore Files list.
4. Press the **Spacebar**. The words "All Files" will appear next to the drive letter C. This tells Backup to restore all the files on the backup disks.
5. Press **Tab** until the **Start Restore** option is highlighted and then press **Enter** to activate it.
6. Insert the backup disks you created in the previous lesson into drive A as prompted and press **Enter** to proceed through the restoration process. (If you

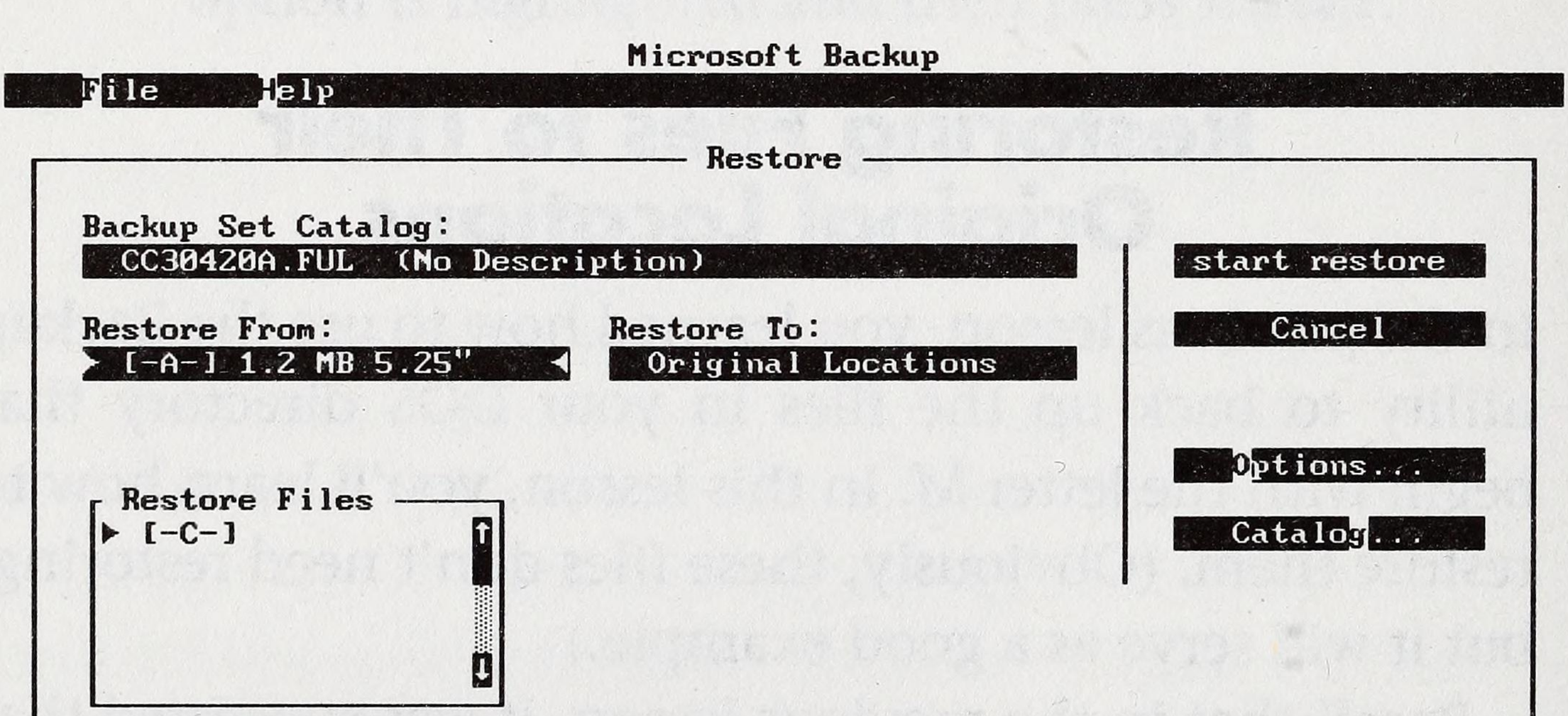


Figure 26.1 The Restore dialog box

used drive B in the previous lesson to create the backup disks, you'll be prompted to insert the floppies in that drive instead.) DOS restores all the files on the backup disks to your C:\DOS directory and then displays a message when the restoration is complete.

7. Press **Enter** to clear the message and return to the main Backup screen.
8. Press **Tab** until the **Quit** option is highlighted and then press **Enter** to leave Backup and return to the DOS prompt.

◆ *Lesson* ◆

27

Copying Entire Directories

The Xcopy Command

You can copy an entire directory and its subdirectories by using the Xcopy command. This command automatically copies all the files in a directory and (optionally) its subdirectories to the target location you specify. Thus, with a single command, you can create a mirror image of a directory and its subdirectories at another location.

Copying a Single Directory

Often, you'll want to create a backup by copying all the files in a directory to a floppy disk. The easiest way to do this is to use the Xcopy command to copy the directory and all its files at once. For instance, try copying a directory called FINAL to a floppy disk:

1. Insert a formatted floppy disk in drive A.
2. Type **xcopy c:\final a:\final** and press **Enter**. (The use of drive letters—c: and a:—in the source

and target parameters for the Xcopy command are required here.) DOS displays the following message:

```
Does FINAL specify a filename
or directory name on the target
(F = file, D = directory)?
```

Tip

When you're specifying a directory, you can avoid this message by adding a backslash (\) to the end of the target directory name, as in a:\final\.

3. Type **d** to specify directory and press **Enter**. DOS creates the FINAL directory on the floppy disk in drive A and then copies the files from C:\FINAL to that directory.

Unless you use the /s switch, Xcopy does *not* copy subdirectories and any files contained in them. See "Copying Subdirectories" later in this lesson.

Copying to a New Directory

If you want, you can specify a new, non-existent directory for the Destination in the XCopy command. For example, using

```
xcopy c:\final a:\back
```

in the previous exercise would create a new directory called OLDFILES on the floppy disk in drive A. Then all the files in the FINAL directory would be copied to the BACK directory.

Copying to An Existing Directory

Although not usually recommended, you can specify an *existing* directory as the Destination in the XCopy command. For instance, imagine that the directory OLDFILES already exists on drive C and contains existing files. If, in the previous exercise you use

```
xcopy c:\final c:\oldfiles
```

then all existing files in the FINAL directory would be copied (added to) to the OLDFILES directory. The result is that the OLDFILES directory contains any existing files and the files copied from FINAL. If the same file, YR1994.TXT for instance, exists in both the Source and Destination directories, then it is written over during the copy operation.

Copying Subdirectories

Using Xcopy to copy subdirectories to a new location requires only that you add the **/s** switch to the Xcopy command. To demonstrate, suppose that the FINAL directory we've been working with contains two subdirectories called MKTG and SALES. Here's how you would copy all of these to a floppy disk on drive A:

1. Insert a formatted floppy disk in drive A.
2. Type **xcopy c:\final a:\final /s** and press **Enter**. DOS displays this message:

```
Does FINAL specify a file name
or directory name on the target
(F = file, D = directory)?
```

3. Type **d** to specify a directory and press **Enter**.

4. View the new directory structure on the floppy disk by typing **tree a:** and pressing **Enter**.

On the floppy disk in drive A, DOS creates a FINAL directory and copies all files in C:\FINAL to it. Within this directory, DOS then creates the MKTG subdirectory and copies all files in C:\FINAL\MKTG to here. Likewise, the SALES subdirectory is created and all files in C:\FINAL\SALES are copied to it.

By default, the Xcopy command will not copy an empty subdirectory unless you use the /e switch along with the /s switch—for example, /s /e.

◆ *Lesson* ◆

28

Working with ASCII Files

What is an ASCII File?

As you may already know, an *ASCII file* is a text file. ASCII stands for American Standard Code for Information Interchange, and is a common file format used by DOS and many other programs.

Tip

You can tell if a file is ASCII by viewing its contents. If you can read the contents—that is, the file doesn't contain smiley-faces or other strange-looking characters—then it's an ASCII file.

Viewing ASCII Files

The Type command allows you to view the contents of an ASCII file on screen. For example, here's how to view the contents of your AUTOEXEC.BAT located in the root C:\:

1. From any directory, enter **type c:\autoexec.bat**.

2. Press **Enter** and DOS scrolls through the entire contents of AUTOEXEC.BAT on the screen. If you need to, you can pause the scrolling by pressing **Ctrl+S** or the **Pause** key from your keyboard. To resume scrolling, press any key except Pause.

Tip

You can also use the MS-DOS Editor to view the contents of ASCII files.

Printing an ASCII File

By using the Print command at the DOS command line, you can print an ASCII file in the background while you're using other DOS commands. For example, try printing the AUTOEXEC.BAT file (an ASCII file discussed in Lesson 30) in your root directory:

1. Move to the root directory C:\ by typing **cd** at the command prompt and pressing **Enter**.
2. Type **print autoexec.bat**. (From any other directory, you'd need to use print c:\autoexec.bat instead.) DOS displays this message asking for the printer location:

```
Name of list device [PRN]:
```

3. Type the **name of the port** to which the printer is connected from these choices—lpt1, lpt2, lpt3, com1, com2, com3, or com4. If you don't specify a port name, DOS will print to PRN (lpt1). In this case,

let's assume the printer is connected to lpt1, so just press **Enter**. You'll now see this message:

```
Resident part of Print loaded

C:\AUTOEXEC.BAT is currently being printed
```

Since DOS prints the file "in the background", you can begin using other DOS commands or run other programs without having to wait for the Print command to complete its task.

◆ *Lesson* ◆

29

Creating Batch Files

Understanding Batch Files

Batch files let you automate operations at the DOS command line. These files are simply ASCII files with a .BAT extension that contain DOS commands. You can run a batch file by typing its name (with or without the .BAT extension) at the DOS prompt. DOS will read the commands in the file and execute them as quickly as your PC will allow. You can create a batch file using any text editor capable of opening and saving ASCII files—for example, the MS-DOS Editor.

Creating a Simple Batch File

In this section, we'll create a simple batch file that makes your C:\DOS directory current and then copies all the .COM files in that directory to drive A. Although this batch file won't be terribly useful, you'll be able to apply the principles you learn when creating your own batch

files. Here's how to create this file from the command line:

1. Insert a formatted floppy disk in drive A.
2. Open the MS-DOS Editor by typing **edit** and pressing **Enter**. Then press **Esc** to clear the opening dialog box.
3. In the Editor, type **c:** and press **Enter**. This line in the batch file will make your C drive current no matter which drive you happen to be using when you run the batch file.
4. Type **cd \dos** and press **Enter**. This line will make the DOS directory current.
5. Type **copy *.com a:** and press **Enter**. This line will copy all .COM files in the DOS directory to drive A. Your screen should now look like Figure 29.1.
6. Save the file to the DOS directory under the name c.bat by choosing the **File Save As** command from the Editor menu. In the text box, type **c:\dos\c.bat** and press **Enter**.
7. Leave the Editor and return to the DOS command line by choosing the **File Exit** command from the Editor menu. Your new batch file is now complete and ready to be run.

Running the Batch File

To run your new batch file from the command line, simply type the name of the file, **C** in this case, and press **Enter**. (No file extension is needed.) Based on the commands shown in Figure 29.1, DOS makes your C drive current, changes to your C:\DOS directory, and begins copying all the .COM files in that directory to the disk in

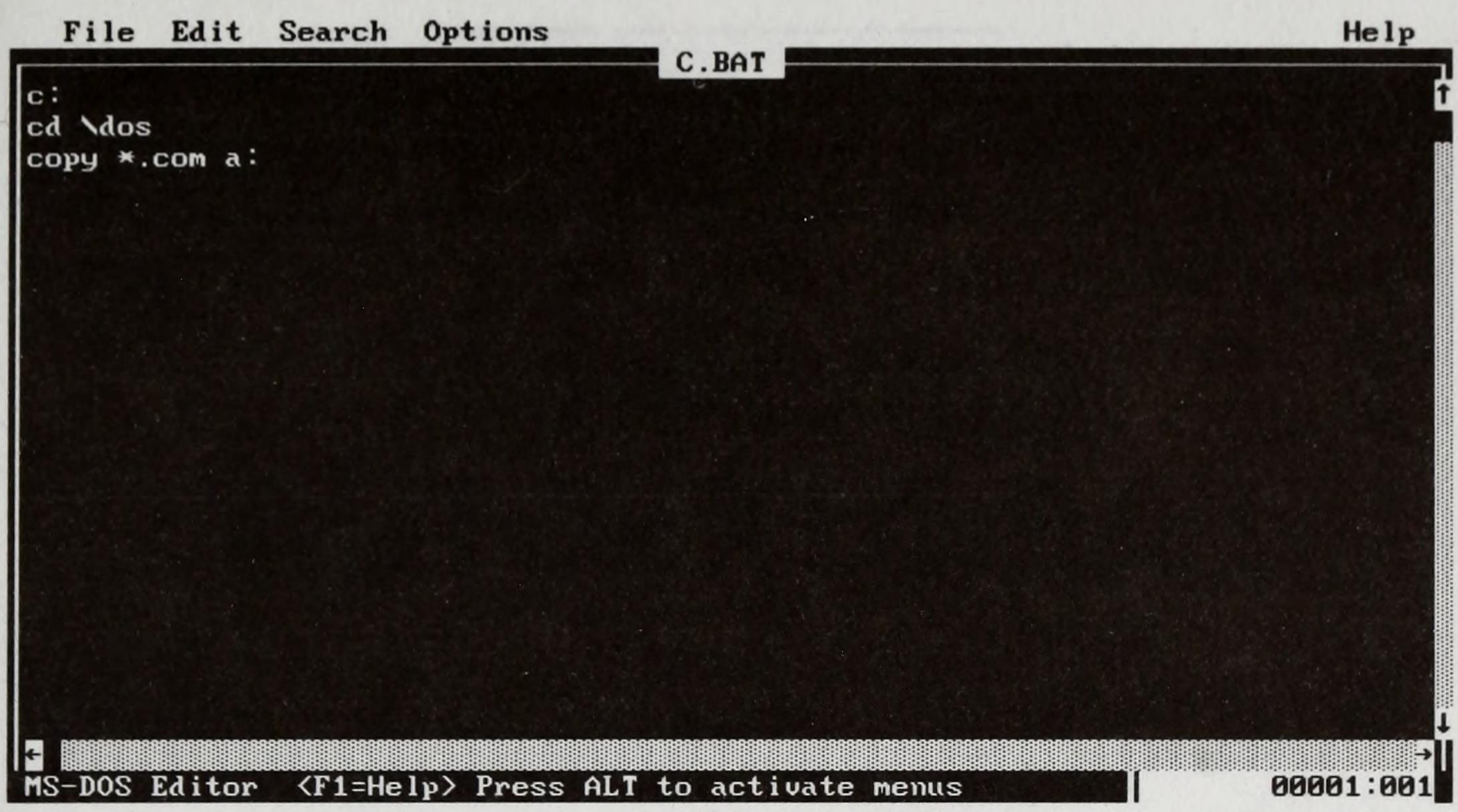


Figure 29.1 A simple batch file

drive A. When the batch file is finished running, you are returned to the DOS prompt.

If you don't like what a batch file is doing, you can stop it at any time by typing **Ctrl+c** or **Ctrl+Break**. DOS will display the message:

```
Terminate batch job (Y/N)?
```

Type **Y** to stop the batch file and return to the DOS prompt.

◆ *Lesson* ◆

30

Modifying AUTOEXEC.BAT

The PC's Boot Sequence

When you start your PC, it performs a series of internal hardware and system checks. Next it checks drives A and B for a bootable disk—a disk with COMMAND.COM and two other hidden DOS system files, IO.SYS and MSDOS.SYS. If a bootable disk is there, your computer boots from that disk (loads the DOS system files into memory from that disk); otherwise, your computer looks to the root directory of drive C for the DOS system files. If these files are found on the root directory of drive C, your PC will boot from that drive instead.

During the boot procedure, DOS loads the file COMMAND.COM, which gives your PC the ability to respond to the routine DOS commands required to manage directories and files. It also automatically looks for two other files—CONFIG.SYS and AUTO-EXEC.BAT. If these files are present, DOS loads them and runs the instructions they contain.

Both CONFIG.SYS and AUTOEXEC.BAT are ASCII text files that can be viewed and edited using the MS-DOS

Editor. They also contain command-line instructions, similar to the ones you might enter at the DOS prompt. The purpose of these command lines is to run different programs that customize the internal configuration and operation of your PC.

The CONFIG.SYS file is always run first. In general, this file contains instructions for loading device drivers (special programs that tell DOS how to interact with your specific equipment). A version of CONFIG.SYS is created automatically by DOS when you install the software on your PC.

The AUTOEXEC.BAT file is run next. While this file can contain a variety of instructions, its primary function is to run the programs and DOS commands you invoke each time you start your computer. Therefore, this is the configuration file most frequently edited by DOS users. You can use AUTOEXEC.BAT do such things as change the appearance of the DOS prompt, or define a sequence of directories in which DOS will automatically search for files. Like CONFIG.SYS, DOS creates a version of AUTOEXEC.BAT when you install the software.

Before You Proceed

In the sections that follow, we'll ask you to load AUTOEXEC.BAT into the MS-DOS Editor to have a look at the file contents. We'll also tell you about different entries you can make in the file to customize the operation of your PC. Before you proceed, however, we *strongly* recommend that you back up AUTOEXEC.BAT and create a system disk (a bootable floppy disk). Both procedures are defined below. That way, if anything goes wrong, you'll be able to reboot your PC and start fresh.

Backing Up AUTOEXEC.BAT

To back up AUTOEXEC.BAT, follow these steps:

1. At the DOS command line, type **c:** and press **Enter** to make drive C current. Then, to make the root directory C:\ current, type **cd ** and press **Enter**.
2. Type **copy autoexec.bat autoexec.bak** and press **Enter**. This creates a second copy of AUTOEXEC.BAT under the name AUTOEXEC.BAK.

Creating a System Disk

To create a system disk (see Lesson 14), you need a blank, formatted floppy disk. When you're ready, follow these steps:

1. Insert the formatted floppy disk in drive A.
2. At the DOS command line, type **cd \dos** and press **Enter** to make the C:\DOS directory current.
3. Type **sys a:** and press **Enter**. A short time later, DOS displays the message:

```
System Transferred
```

IF SOMETHING GOES WRONG

If something goes wrong in the example that follows, and your PC doesn't work the way it used to, or you cannot reboot your PC, perform the following steps:

1. Insert the system disk you created in the previous section into drive A.

2. Press **Ctrl+Alt+Del**. Your computer will restart and DOS will prompt you for the date and time. Press **Enter** in response to both of these prompts. The DOS prompt **A:>** will then be displayed.
3. Type **c:** and press **Enter** to make drive C current.
4. Type **copy autoexec.bak autoexec.bat** and press **Enter**.
5. Remove the system disk from drive A and press **Ctrl+Alt+Del** to reboot your computer.

Loading AUTOEXEC.BAT into MS-DOS Editor

You can take a look at AUTOEXEC.BAT by loading it into the MS-DOS Editor. Here's how to do this:

1. From the DOS command line, open the MS-DOS Editor by typing **edit** and pressing **Enter**. Then press **Esc** to clear the opening dialog box.

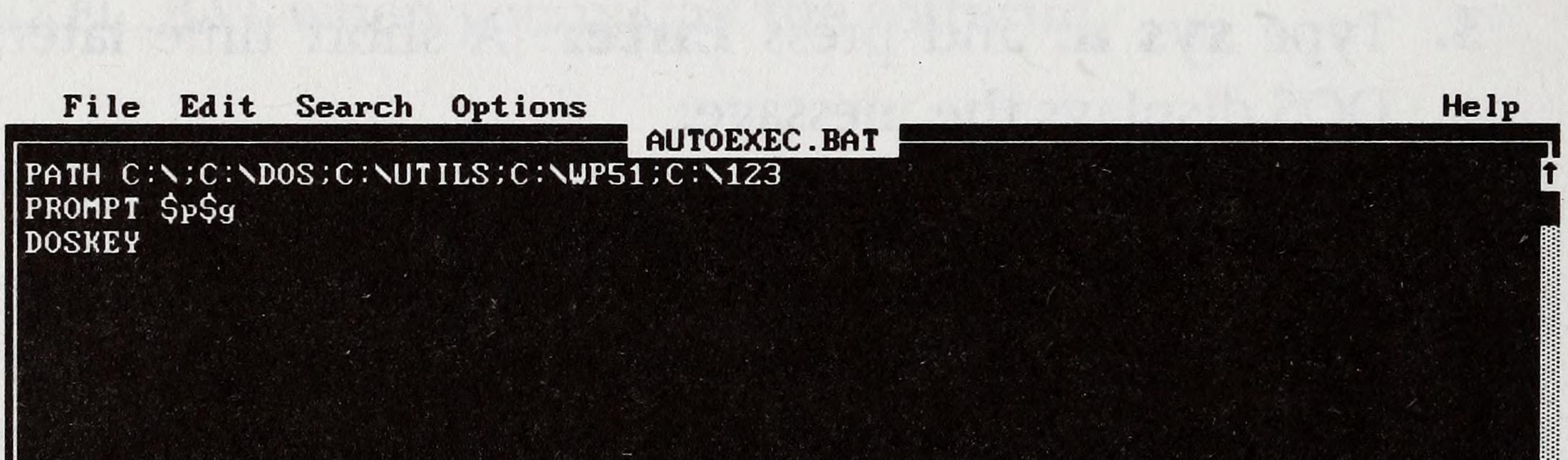


Figure 30.1 A typical AUTOEXEC.BAT file

2. Load AUTOEXEC.BAT into the Editor by choosing the **File Open** command from the Editor menu. Then type **c:\autoexec.bat** in the File Open dialog box. Press **Enter** and the AUTOEXEC.BAT file, similar to the one in Figure 30.1, appears on your screen. (Your file, naturally may be different than that shown in Figure 30.1.)

AUTOEXEC.BAT Commands

There are three commands in AUTOEXEC.BAT that any serious DOS user should know about. They are:

Path	Defines a series of directories in which DOS will search for program files.
Prompt	Configures the appearance of the DOS prompt.
Doskey	Loads the Doskey program that allows you to easily repeat commands recently entered at the DOS command line.

The sections that follow give a brief overview of how these commands are used.

The Path Command

The Path command is used to define a series of directories through which DOS will search for program files (.EXE, .COM, or .BAT files). When you run a program, DOS will always look in the current directory first. After that, DOS looks in the directories defined by the Path command in AUTOEXEC.BAT. As long as the program you want to run

is included in one of these directories, DOS will run it, regardless of which directory happens to be current. This means you can run a program from any directory just by typing its file name at the command line and pressing Enter; you don't have to include the full path to the program file.

To create a Path statement, type **PATH** followed by a space and the drive and directory path of each of the directories that you want DOS to search. Separate your entries with semicolons(;). DOS will search the directories in the order in which they are listed. For example, the AUTOEXEC.BAT file in Figure 30.1 includes the following Path command statement:

```
PATH C:\;C:\DOS;C:\UTILS;C:\WP51;C:\123
```

With this Path statement, DOS will first search the root directory (\) on drive C, followed by C:\DOS, C:\UTILS, C:\WP51, and C:\123, in that order. If it can't find the program you are trying to run in any of these directories, DOS displays an error message.

If a directory containing a program that you run frequently is not included in your Path statement, you can easily add it. To do this, move the cursor to the end of the Path statement, type **;** (semicolon) followed by the drive and path to the directory.

⇨ ***Note:*** When you install a new program, its Installation program will, in many cases, automatically attempt to update the Path statement in AUTOEXEC.BAT to include the directory in which the program has been installed.

The Prompt Command

Unless you specify otherwise, the DOS prompt consists of the current drive letter followed by a greater than (>) symbol, for example, **C>**. The Prompt command, however, lets you control the appearance of the DOS prompt.

Probably you already have a Prompt statement in your AUTOEXEC.BAT file, but in case you don't, we recommend you include the following statement on a separate line:

```
prompt $p$g
```

This particular version of the Prompt command has two parameters: **$p** and **$g**; **$p** causes the path to the current directory to be displayed as part of the DOS prompt. Without this, it's very difficult to tell which directory you're in when working at the DOS command line. The **$g** parameter causes a greater than (>) symbol to appear at the end of the DOS prompt. With this in mind, if C:\MYFILES is the current directory, your DOS prompt would look like this:

```
C:\MYFILES>
```

The Doskey Command

The Doskey command runs the Doskey utility (DOSKEY.COM) located in your C:\DOS directory. This program records the commands you've entered recently and allows you to quickly recall and rerun them. (Refer to Lesson 17 for details on how to work with this utility.) If

the Doskey command is not included in your AUTOEXEC.BAT file, you can easily include it by typing **doskey** on a separate line.

Saving AUTOEXEC.BAT in the MS-DOS Editor

After you've made any changes to your AUTOEXEC.BAT file in the MS-DOS Editor, you must save them to disk. In order for these changes to take effect, you must then reboot. Here's how to do this:

1. Select the **File Save** command from the Editor menu. This saves the file under its existing name AUTOEXEC.BAT.
2. Leave the Editor by choosing **File Exit** from the Editor menu. you'll be returned to the DOS command line.
3. Use the changes you made to AUTOEXEC.BAT during the current session by pressing **Ctrl+Alt+Del** to reboot your computer. Otherwise, the modified AUTOEXEC.BAT file won't take effect until the next time you turn on your PC.

◆ *Lesson* ◆

31

Increasing Disk Space Using DoubleSpace

Compressing Your Hard Disk

If you're using DOS 6, you can nearly double the size of your hard disk through disk compression. DoubleSpace is the utility that performs the compression and then maintains it on an ongoing basis.

⇨ ***Note:*** Many people set up DoubleSpace immediately after installing DOS 6. You can check to see if this has already been done to your PC by using the **dir /c** command at the DOS command prompt. If you see any disk compression information as part of the normal directory listing, you can tell that DoubleSpace is already active, and you can skip the first part of this lesson.

One of the best things about DoubleSpace is that you don't need to understand how it works to benefit from its use. And because a compressed drive works like an uncompressed drive, you don't have to change any of your work habits.

Setting Up DoubleSpace

Before you can gain the benefits of DoubleSpace, you must first set it up. Here are the steps:

1. Quit all running programs, including the DOS Shell.
2. Type **dblspace** at the DOS prompt and then press **Enter**.
3. Press **Enter** again to continue past the initial welcome screen.
4. When DOS shows the screen in Figure 31.1, press **Enter** to accept Express Setup. You'll then see a screen telling you how long it will take to compress your hard drive.
5. Press **C** to continue. DoubleSpace will reboot your PC and then take the time it needs (several minutes to several hours depending on the amount of information you have stored) to compress your drive. When the compression process is complete, you'll see a screen like the one in Figure 31.2 showing how

```
Microsoft DoubleSpace Setup

    There are two ways to run Setup:

    Use Express Setup if you want DoubleSpace Setup to compress
    drive C and determine the compression settings for you. This
    is the easiest way to install DoubleSpace.

    Use Custom Setup if you are an experienced user and want to
    specify the compression settings and drive configuration
    yourself.

    +-----------------------------------------------------------+
    | Express Setup (recommended)                               |
    | Custom Setup                                              |
    +-----------------------------------------------------------+

    To accept the selection, press ENTER.

    To change the selection, press the UP or DOWN ARROW key
    until the item you want is selected, and then press ENTER.

ENTER=Continue  F1=Help  F3=Exit
```

Figure 31.1 Choosing Express Setup

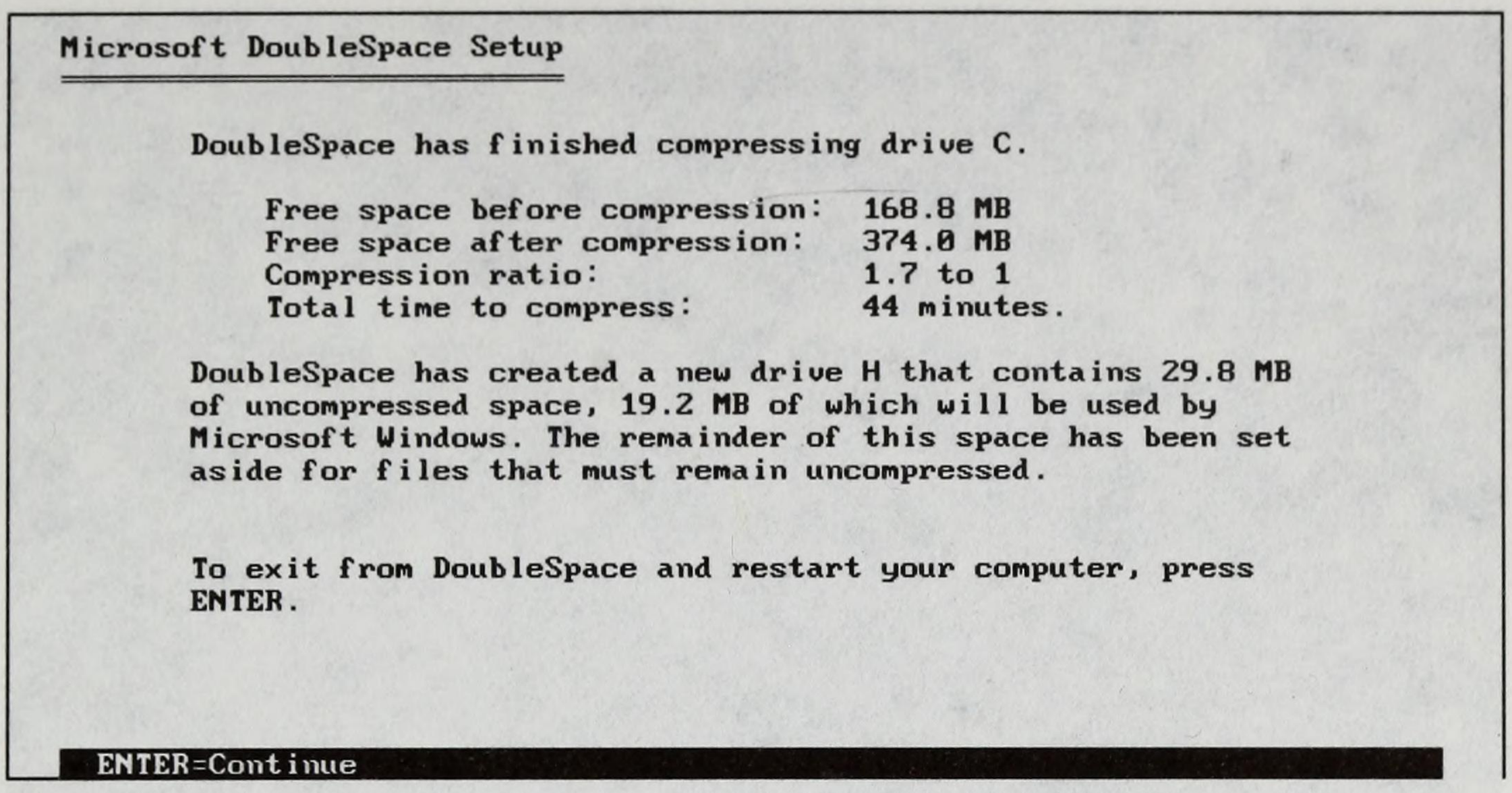


Figure 31.2 After DoubleSpace has compressed drive C

long the process took and the amount of free space on your hard disk both before and after compression.

6. Press **Enter** to exit DoubleSpace and have DOS reboot your computer a second time.

You can now use your compressed drive in the same way you did before it was compressed.

Seeing the Effects of DoubleSpace

Try this procedure to see the effects DoubleSpace has had on your hard drive:

1. Type **dir /c** at the DOS command prompt. Along with the normal directory listing, you'll see an extra column showing compression information for each file listed. Also, at the end of the directory list, you'll

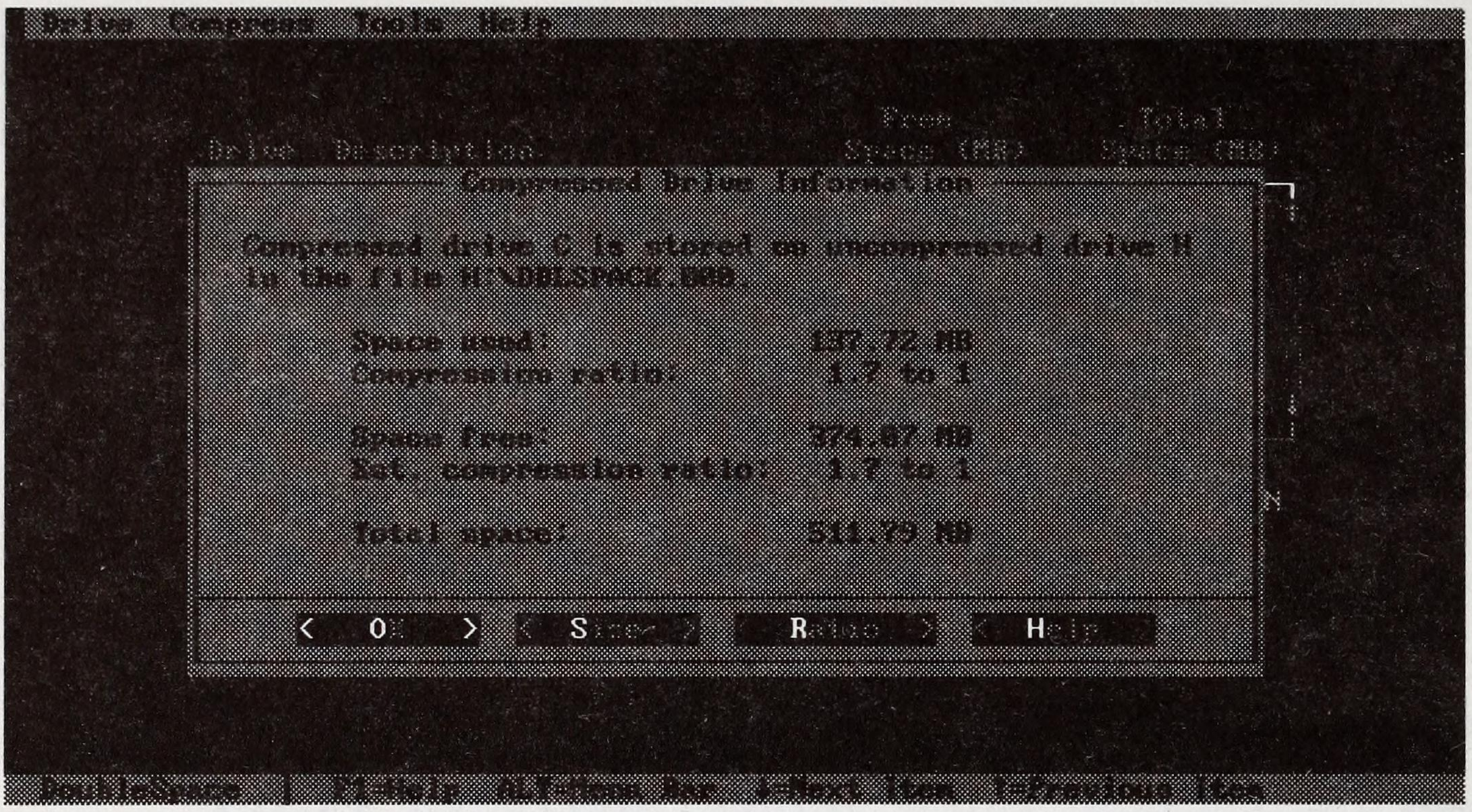


Figure 31.3 DoubleSpace's detailed drive information

see the average compression ratio for the listed files in a form like this:

```
3.0 to 1.0 compression ratio
```

2. Type **dblspace** at the DOS prompt to start the DoubleSpace utility. You'll then see a screen showing the compressed drives on your PC.
3. Press **Enter** to get more detailed information about drive C. You'll then see a screen like the one in Figure 31.3.
4. Press **Enter** to clear the detailed drive information.
5. Select the **Drive Exit** command (Alt, D, X) to leave DoubleSpace and return to the DOS prompt.

◆ *Lesson* ◆

32

Defragmenting Your Hard Disk

The Problem with Fragmentation

The more you use your PC, the more the information on your hard disk can become fragmented. This occurs naturally because files are written to the disk drive in small fixed-sized chunks rather than as complete units. Fragmentation occurs when the chunks that make up a file are scattered across various locations on the hard disk. This scattering doesn't affect the information in the file, but it does take your PC longer to access the file.

Combatting Fragmentation with Defragmenter

If you're running DOS 6, you can combat file fragmentation by periodically running Defragmenter. Defragmenter eliminates fragmentation by physically rearranging the files on your hard disk, thus increasing disk access speed.

Before running Defragmenter, though, it's wise to do some housekeeping first to delete any incomplete files that might reside on your hard disk. Perform these steps:

1. Quit all running programs, including the DOS Shell.
2. Type **chkdsk /f** at the DOS prompt. This checks for any incomplete files (so-called "lost allocation units") that were not fully written to disk due to power outages or program crashes. If any such files exist, you'll see a message like this:

```
20 lost allocation units found in 5 chains.
Convert lost chains to files?
```

3. Press **Y** to save these lost allocation units to files that can be deleted. The files will be written to the root directory of your hard disk as FILE0001.CHK, FILE0002.CHK, and so on.
4. Type **cd ** and press **Enter** to move to the root directory of your hard disk.
5. Type **del file*.chk** and press **Enter** to delete the lost-chain files.

The space previously occupied by the lost-chain files is now recovered. Now you can perform the following steps to run Defragmenter:

1. Type **defrag** at the DOS prompt and press **Enter**. Figure 32.1 shows the screen that appears listing the drives available for defragmenting.
2. Press **Enter** to select drive C. Following a brief pause, Defragmenter will recommend a defragmentation option.

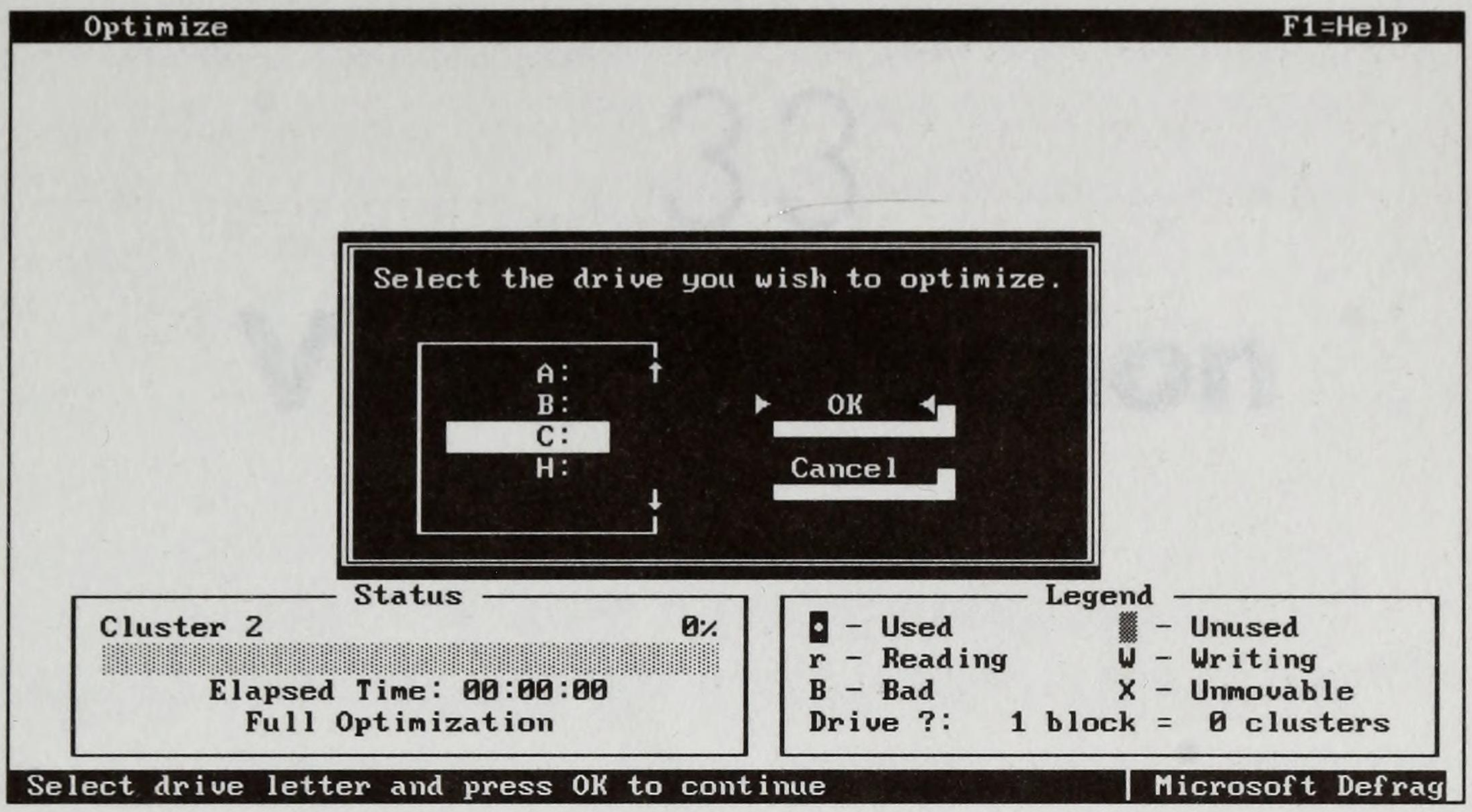


Figure 32.1 Drives available for defragmenting

Depending on the size of your hard disk and the nature of its fragmentation, Defragmenter may require a substantial amount of time to defragment your drive. Be prepared to leave your PC idle for an extended period.

3. Press **Enter** to accept the suggested option and start the defragmentation process. Defragmenter will display a message when the defragmentation is complete.
4. To clear the message and leave the Defragmenter, press **Enter** to and then press **Alt+x** to select the Exit command.

◆ *Lesson* ◆

33

Virus Detection

Protecting Your System from Viruses

If you're using DOS 6, you can protect your system from over 800 known viruses by periodically running the Anti-Virus utility. This utility scans all the files on your hard disk and removes any viruses it encounters.

What are Viruses?

Viruses are programs meant to spread throughout the PC community, often causing severe damage in the process. There are thousands of known viruses, but the most well known is the Michelangelo virus, which lurks in your system until March 6 and then destroys all the data on your hard disk when you start up your PC on that day.

Viruses actually fall into three categories:

- Boot-sector virus—The *boot sector* is a physical location on your hard disk from which DOS is loaded into RAM when you start your PC. A boot-sector

virus—for example, the Michelangelo virus—replaces the code in the boot sector with its own code, allowing the virus to load itself into memory as DOS is loaded. This type of virus is meant to infect your system without your knowing.

- Trojan horse—This is an actual program that looks inviting until you copy it to your hard disk and then execute it. At this point, the program turns malicious, slowing down your system, destroying files, or, in some cases, trashing all the data on your hard disk.
- File infector—This is the most common type of virus and attaches itself to program files. Then, when you execute an infected program file, the virus spreads to other program files.

Using Anti-Virus to Scan for Viruses

Although viruses have gained much notoriety, the chances of infection are slim. You can easily protect your system from infection by running DOS 6's Anti-Virus utility whenever you install a new program or read a floppy disk given to you by someone else.

Here are the steps for running Anti-Virus:

1. Type **msav** at the DOS prompt. Figure 33.1 shows the screen that appears.

If Anti-Virus doesn't start, chances are it wasn't installed on your system when DOS 6 was installed. See "Installing Anti-Virus, Backup, and Undelete After Setup" in Chapter 1 of the *MS-DOS 6 User's Guide*.

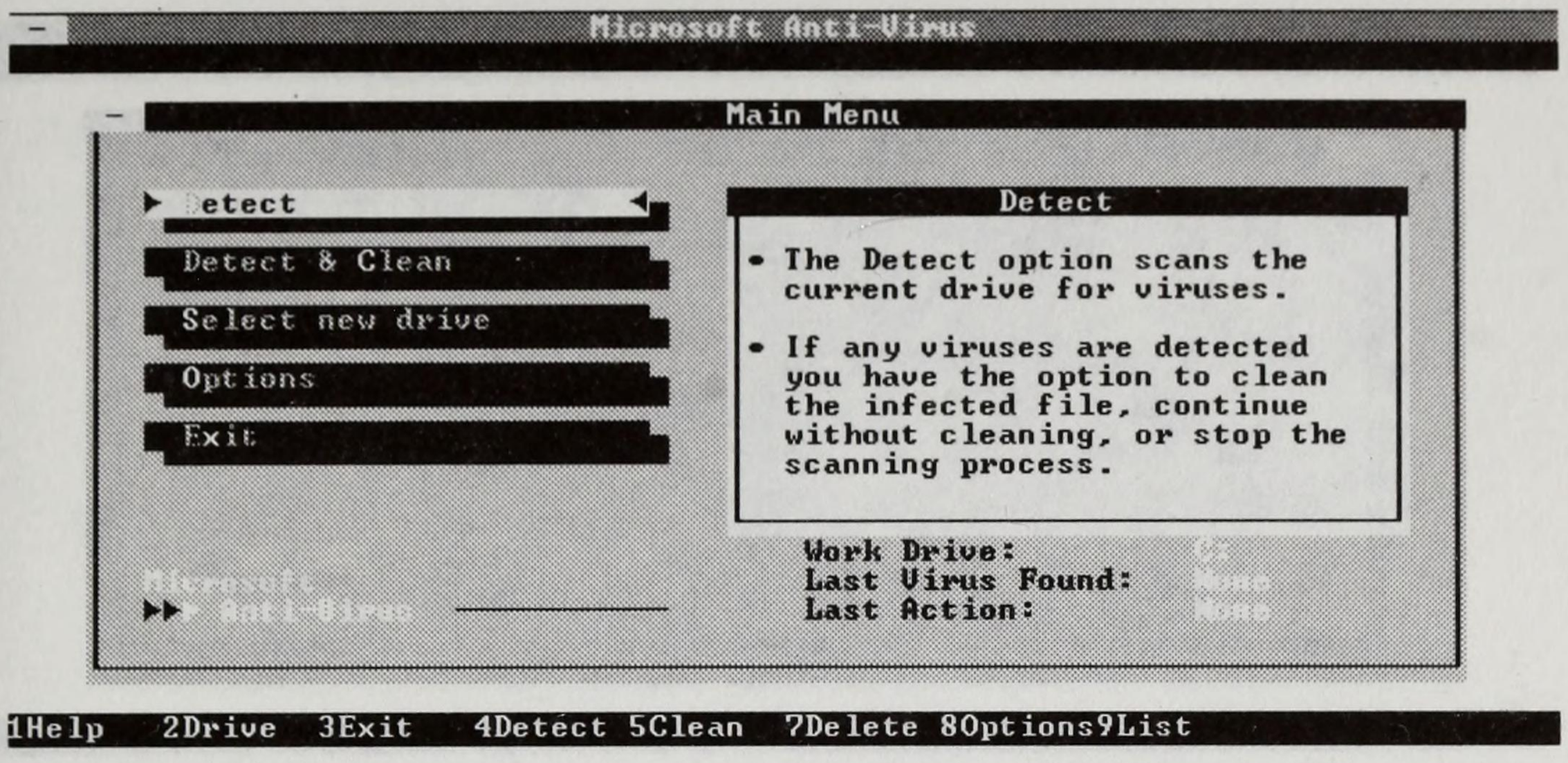


Figure 33.1 The Anti-Virus main screen

2. Press **Enter** to select the Detect option and have Anti-Virus start scanning drive C. If a virus is encountered, you'll see a message similar to the one in Figure 33.2.
3. Press **Enter** to select Clean and remove the virus from your system. When Anti-Virus has completed the scan, you'll see a screen like the one in Figure 33.3.
4. Press **Enter** to clear the results and return to the main screen.
5. To return to the DOS prompt, press **Alt+x** to choose the Exit command and then press **Enter** to clear the final confirmation message.

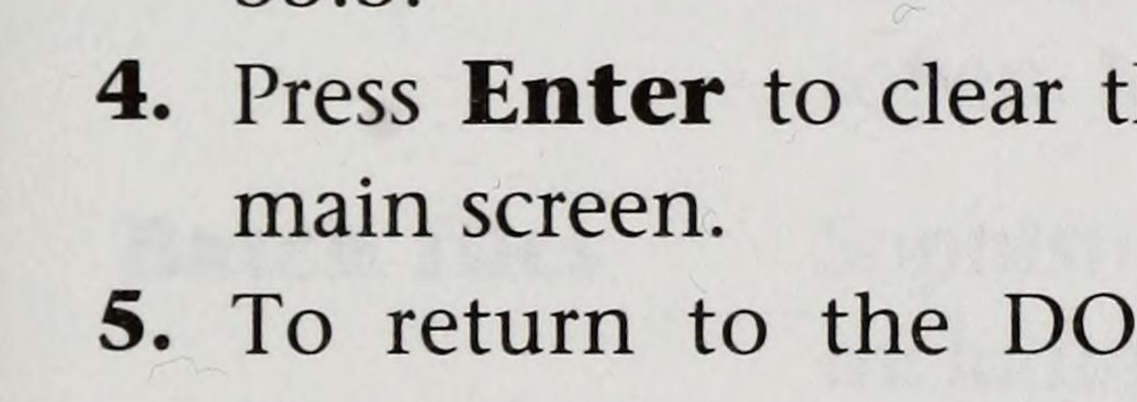


Figure 33.2 When Anti-Virus finds a virus

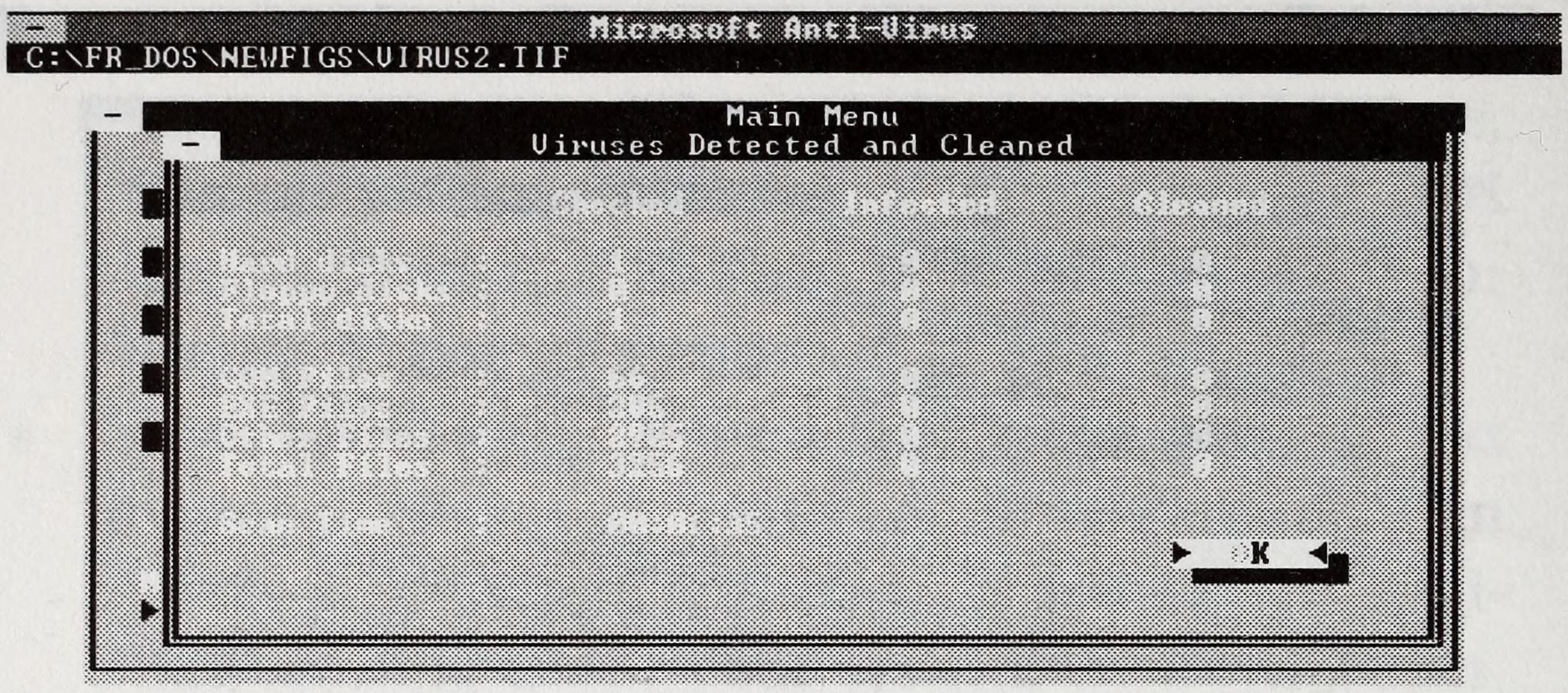


Figure 33.3 Results of the scan

Epilogue
Where To Go from Here

Obviously, this book doesn't tell you everything there is to know about DOS. The following are suggested areas in which you may want to do additional research:

Doskey — Editing command lines and creating macros.

Commands — Important DOS commands that have not been discussed in this book, including Append, Chkdsk, Debug, Fastopen, Mem, and Unformat.

Batch files — Sophisticated batch file techniques, including branching, looping, pausing, and having one batch file call another.

Optimization — Optimizing memory, setting up disk cache, and setting up a RAM disk.

Configuration	Customizing AUTOEXEC.BAT and CONFIG.SYS, configuring your ports, and modifying your screen and keyboard.
Disk Management	Repartitioning and reformatting a disk drive, and speeding up your disk drive using SMARTDrive.
Command Techniques	Redirection, piping, and filters.
Variables	Environment variables—for example, Path, Comspec, Shell, and Temp.
QBasic	Writing your own custom BASIC programs.

Here are some more areas to investigate that are unique to DOS 6:

Memory Management	Making more memory available to your programs using MemMaker.
Configuration	Starting your PC with a choice of configurations.
Windows	Using the Windows-based versions of Backup, Anti-Virus, and Undelete.
Diagnostics	Using the Microsoft Diagnostics Program (MSD) to get detailed technical information about your PC's hardware, device drivers, and interrupts.

Disk Management	Fine tuning disk compression and defragmentation.
Linking PCs	Connecting two computers using a serial cable and the Interlnk utility.
Virus Protection	Using Vsafe, a memory resident program, to monitor for viruses on an ongoing basis.

Index

A

Active Task List, 113, 114, 115
Archive file, 87
Arrow keys, 9
ASCII files, 38
  editing of, 93
  printing of, 186–187
  viewing of, 185–186
AUTOEXEC.BAT, 152
  back up of, 195
  and boot sequence, 194
  Doskey command, 199–200
  loading into MS-DOS Editor, 196–197
  Path command, 197–198
  Prompt command, 199

B

Backing up files, 169–176
  Backup Utility, 170–173
  in DOS directory, 173–176
  importance of, 169
Backslash (\), 128–129
BACKUP.EXE, 45, 70, 153
Backup Utility, 70, 170–173
  configuration sequence, 170–173
  uses of, 170
.BAS files, 38, 83
.BAT files, nature of, 37–38
Batch files
  creation of, 189–190
  functions of, 189
  running of, 190–191
Boot disk, creation of, 104–105, 195–196
Boot sequence, 193–194

C

Canceling commands, 123
Cd, 128–129

Chdir, 158
Check box, 8
Check Disk, 77
Check Mem, 75, 77–78
Colors, DOS Shell,
    changing of, 79–80
    nature of, 37
COMMAND.COM, 44, 70,
    104, 193
    and boot sequence,
    193–194
    purpose of, 120
Command line
    help at, 125–126
    running programs from,
    151–152
    running Utilities from,
    153
Command line commands
    copying files, 142–145
    deleting files, 145–146
    moving files, 146–149
    renaming files, 141–142
Command line shortcuts,
    155–158
    and current directory,
    156–157
    and current drive,
    155–156
    subdirectory shortcut, 157
Command name, 121
Command Reference,
    viewing, 40
Commands
    canceling commands,
    123
    Doskey, use of, 124–125
    entering commands,
    120–121
    parts of, 121
    pausing commands,
    122–123
    repeating last
    command, 123–124
    syntax for, 121–122
Compression, of hard
    disk, 201
CONFIG.SYS, and boot
    sequence, 194
Copy, 142–145
Copying directories,
    181–183
    to existing directory, 183
    to new directory, 182
    single directory, 181–182
    Xcopy command, 181
Copying disks, 107–110
    Disk Copy, steps in use,
    107–110
Copying files, 22–28
    to another directory,
    24–26
    to another drive, 145
    changing names while
    copying, 143–144
    from command line,
    142–145
    between drives, 27–28
    file selection, quick
    method, 24
    group of files, 144–145
    with mouse, 25, 27–28
    single file, 143
    within a directory, 23–24

Copying subdirectories, 183–184
Ctrl+Alt+Del, 200

D

Data files, 83
Date, sorting files by, 63
Defragmentation, 205–207
  steps in use, 206–207
Del, 145–146
Delete protection
  Delete Sentry, 163–165
  Delete Tracker, 163–166
Delete Sentry, 163–165
  enabling protection of, 165
  operation of, 164
Delete Tracker, 163–166
  enabling protection of, 166
  operation of, 164–165
Deleting files, 35
  from command line, 145–146
  from disk, 35
  recovery of, 35
Deleting program items, 76
Del key, 9
Deltree, 19, 132–133
Destination disk, and copying disks, 109, 110
Dialog boxes, DOS shell, 7–9
  check box, 8
  editing keys, 9
  moving among items, 8–9
  push button, 8
  radio buttons, 7
  text box, 7
Dir, 131
Directories
  changing directories, 13–16, 128–129
  coping entire directory, 181–184
  copying files to another directory, 24–26
  copying files within, 23–24
  creating new directory, 18, 129–130
  current directory, 13–14, 156–157
  directory information settings, 57
  dot (.), 157
  functions of, 2
  parent directory, 12
  path, 17–18
  referring to last used directory, 158
  removing directories, 18–19, 131–133
  root directory, 11
  structure of, 11–12
  subdirectory, 11–12
  and undelete, 159
  viewing contents of, 129
Directory tree
  keys in, 15
  navigation of, 14
  viewing of, 130–131

Directory tree area, DOS Shell screen, 4
DiskCopy, 70
steps in use, 107–110
DISKCOPY.COM, 45, 70, 153
Disk operating system. *See* DOS
DiskOrder, sorting files by, 63
Disks
copying disks, 107–110
disk information settings, 58
formatting, 99–105
Disk Utilities group, accessing, 44–45
Display adapter, 81
Display mode
display adapter, 81
graphics mode, 81–82
resolution settings, 81
text mode, 80–81
DOS, functions of, 1–2
Doskey, 199–200
adding to AUTOEXEC.BAT, 125
function of, 199
use of, 124–125
DOS Shell, 3–9
changing screen arrangement, 53–54
colors, changing of, 79–80
dialog boxes, 7–9
display mode, changing of, 80–82
exiting, 9
file associations, changing of, 83–84
menu commands, selection of, 5–7
programs/commands, 153
purpose of, 119
returning to, 41
running programs from, 39–40
screen areas in, 3–5
updating, 64
Dot (.), 157
DoubleSpace, 202–204
seeing effects of, 203–204
set up steps, 202–203
Drag-and-drop method, with mouse, 25
Drive Exit, 204
Drive icon areas, DOS Shell screen, 3–4, 5
Drives
changing drives, 12–13, 127
coping files to different drive, 145
copying files between, 27–28
current drive, 12, 155–156
Dual File Lists view, 26, 31

E

EDIT.COM, 44, 70, 151, 153

End key, 9, 15
.EXE files, 84, 112
  nature of, 37
Extensions. *See* File extensions

F
File Allocation Table (FAT), 164
File Associate, 38
File associations
  changing of, 83–84
  functions of, 83
  running of, 85
File attributes
  archive file, 87
  assignment of read only attributes to, 88
  changing for group of files, 89
  changing of, 87–88, 139
  hidden files, 87
  read only file, 87
  system files, 88
  viewing of, 138
File Change Attributes, 88
File Copy, 22–24, 48
File Create Directory, 18
File Delete, 35
File Deselect All, 51
File display, 59–64
  of hidden files, 62
  limiting types of files displayed, 59–60
  sort order, changing of, 62–64
  and wildcard matches, 61
File Exit, 9, 40, 41, 97, 116, 126, 151, 152, 200
File extensions
  reasons for use, 22
  sorting files by, 63
File information, 135–139
  about group of files, 136
  about specific file, 135
File list, control of, 137–138
File list area
  DOS Shell screen, 4
  title bar, 18
File Move, 29–31
File name
  sorting files by, 63
  See also Naming files
File New, 74–75
File Open, 30, 39
File Print, 97
File Properties, 77
File Rename, 33–35
File Reorder, 76
File Run, 40–41
Files
  backing up files, 169–176
  copying files, 22–28
  deleting files, 35, 145–146
  file display, 59–64
  file information settings, 57
  moving files, 29–32
  naming files, 21–22

Files *(continued)*
nature of, 1–2
renaming files, 32–35
restoring files, 177–179
searching for files, 65–68
selection of, 47–51
types run from DOS, 37–38
undeleting, 159–162
undeleting files, 159–162
viewing contents of, 89–91
viewing files, 53–56
File Save, 200
File Save As, 95, 190
File Search, 65–66
File Select All, 51
File specifier, 59
and wildcards, 61
File View File Contents, 90–91
Format, 70–71, 100–103
FORMAT.COM, 45, 70–71, 153, 154
Formatting disks, 99–105
quick format, 103–104
steps in, 100–103
testing disk for format, 99–100

G

Graphics mode, 81–82

H

Hard disk
compression of, 201
defragmentation of, 205–207
DoubleSpace, 202–204
HELP.COM, 39, 42
Hidden files, 87
display of, 62
Home key, 9, 15

I

IO.SYS, 62, 193

M

MEM.EXE, 77
Memos, 94–97
creating new memo, 94–95
printing memo, 96–97
saving, 95–96
Mkdir, 129–130
Mouse
copying files, 25, 27–28
drag-and-drop method, 25
dragging, 15
moving files, 31
running program with, 40
use in DOS Shell, 7
Move, 30, 146–149
Moving files, 29–32
to another directory, 31
from command line, 146–149
from DOS Shell, 29–30
and duplicate filenames, 32
group of files, 148–149

from keyboard, 29
with mouse, 31
and renaming files, 32–33, 148
single file, 147
MS-DOS Editor
ASCII files, editing of, 93
loading AUTOEXEC.BAT into, 196–197
memos, creation of, 94–97
running from keyboard, 42–43
start up of, 70
MSDOS.SYS, 62, 193

N

Naming files, 21–22
file extensions, 22
filename rules, 21
prohibited characters in, 21

O

Options Colors, 5–6, 79–80
Options Display, 81–82
Options Enable Task Swapper, 111–112
Options File Display Options, 60, 62, 63
Options Select Across Directories, 48
Options Show Information, 56
OS2.TXT, 29, 30

P

Page Down key, 15
Page Up key, 15
Parameters, 121
passing to programs, 152
Parent directory, 12
Path, 17–18
creating path statement, 198
function of, 17
viewing of, 17–18
Path command, 197–198
function of, 197
Pausing commands, 122–123
PCTRACKR.DEL, 164, 165, 166
Printing
ASCII files, 186–187
memo, 96–97
Program groups
activating area, 71–72
adding program items to group, 72–75
changing properties of item, 77–78
contents of, 70–71
creating new group, 72
deleting program item, 76
Disk Utilities group, 44–45
Main program group, 44
operations performed with, 69
rearranging groups and items, 76

Program list area
activation of, 71–72, 74
DOS Shell screen, 5
Programs
functions of, 2
locking of programs, 117
passing parameters to, 152
quitting programs, 115–117
running programs, 39–45
starting more than one program, 112–114
Prompt, 2–3
Prompt command, 199
parameters of, 199
Push button, 8

Q

QBasic
programs in QBasic, 38
start up of, 70
QBASIC.EXE, 44, 70
Quick Format, 103–104
Quitting programs, 115–117

R

Radio buttons, 7
Random access memory (RAM), 2
Read only file, 87
Rebooting, 200
Rename, 141–142
Renaming files, 32–35
from command line, 141–142
while copying files, 143–144
in current location, 33–35
and moving files, 32–33, 148
overwriting existing files, 35
Resolution settings, display mode, 81
Restore, 70
RESTORE.EXE, 45, 70, 153
Restoring files, 177–179
steps in use, 178–179
Rmdir, 131–132
Root directory, 11
Running programs, 39–45
from command line, 151–154
from DOS Shell, 39–40
with File Run command, 40–41
from program list area, 42–45

S

Saving, memos, 95–96
Searching for files, 65–68
for group of files, 66–68
for single file, 65–66
Search path, 42
function of, 42
Selection of files, 47–51
all files in directory, 51
consecutive files, 47–49

nonconsecutive files, 50–51
Shift key, 9
Size of file, sorting files by, 63
Sorting files, 62–64
  changing sort order, 63
  refraining from sorting, 63–64
Source disk, and copying disks, 109
Spacebar, as toggle key, 51, 88
Subdirectories, 11–12
  copying of, 183–184
  creating subdirectories, 129–130
  hiding subdirectories, 16–17
  shortcut to referencing of, 157
  viewing subdirectories, 16
Switches, 121
  to control file list, 137–138
  /p switch, 137
  /w switch, 137
Syntax, for commands, 121–122
System disk, creation of, 104–105, 195–196
System files, 88

T
Task switching, 111–117
  quitting programs, 115–117
  starting more than one program, 112–114
  switching programs, 114–115
  turning on and off, 111–112
Text box, 7
Text files. *See* ASCII files
Text mode, 80–81
.TXT files, 38, 83

U
Undelete, 35, 40, 71, 159–162
  in current directory, 160–162
  and directories, 159
  outside of current directory, 162
  parameters for, 166–167
  steps in use of, 160–161
  UNDELETE.INI file, 167–168
  and wildcards, 162
UNDELETE.EXE, 45, 71, 153
Utilities, running from command line, 153

V
View All Files, 55–56
View Dual File Lists, 54

Viewing directory contents, 129
Viewing files, 53–56
  and all files view, 55–56
  and dual file list view, 54
  viewing contents of files, 89–91
View Program/File Lists, 53–54
View Program List, 53
View Refresh, 64
View Single File List, 53
Virus detection, 209–212
  Anti-Virus, running of, 210–211
  boot-sector virus, 209–210
  file infector, 210
  Trojan horse, 210

W

Wildcards
  copying groups of files, 144–145
  in file list area title bar, 18
  and file specifier, 61
  and undelete, 162

X

X copy, 181, 183
  copying directories, 181–183
  copying subdirectories, 183–184

DOS 6 EASY REFERENCE

When using the DOS Shell in DOS 6, there are a few commands and procedures that you will want to have available at all times. With these skills and a little practice, you will be able to accomplish quite a bit.

1. Start the DOS Shell	At the DOS prompt, type DOSSHELL.
2. Move between the four main areas (drive icons, directory tree, file list, main)	Press Tab.
3. Move back to an area	Press Shift + Tab. Or move your mouse pointer to an area's title bar and click your left mouse button.
4. Use the menu bar	Press Alt or F10.
5. Select a menu option	Use the up and down arrow keys to move between options. Press Enter to select an option. Or move your mouse pointer to the option and click your left mouse button.
6. Move between directories in the directory-tree area, files in the file-list area, or programs in the main area	Use the up and down arrow keys.
7. Exit the DOS Shell	Press Alt+F4.

Program
activa
DOS S
Program
functi
lockin
passin
15
quittin
11
runnin
39
startin
pr
Prompt, 2
Prompt c
parame
Push butt

Q

QBasic
progran
start up
QBASIC.E
Quick For
Quitting p
115

R

Radio butt
Random a
(RA
Read only
Rebooting,
Rename, 1
Renaming

Viewing directory contents, 129
Viewing files, 53–56
and all files view, 55–56
and dual file list view, 54
viewing contents of files, 89–91
View Program/File Lists, 53–54
View Program List, 53
View Refresh, 64
View Single File List, 53
Virus detection, 209–212
Anti-Virus, running of, 210–211
boot-sector virus, 209–210
file infector, 210
Trojan horse, 2

W

Wildcards
copying group 144–145
in file list area 18
and file specifi
and undelete,

X

X copy, 181, 183
copying direct 181–183
copying subdi 183–184

DOS 6 EASY REFERENCE

As you begin to use DOS 6, you will want to know the basic procedures for working with files, directories, and disks. The following chart shows how to accomplish this within the DOS Shell and from the DOS prompt.

DOS Shell Action	DOS Command	Result
1. Select Help from menu bar	HELP	Accesses help
2. Select Format from main area Type A:	FORMAT A:	Formats floppy disk in drive A:
3. Select File from menu bar Select "Create Directory" Type TEST	MD TEST	Creates a new directory named TEST
4. Select directory TEST in directory-tree area Select File from menu bar Select "Delete"	RD TEST	Deletes a directory named TEST
5. Select C: in drive-icons area Select TEST in directory-tree area	CD \ CD TEST	Goes to the root directory Goes to a directory named TEST
6. Select file FILE1 in file-list area Select File from menu bar Select "Delete"	DEL FILE1	Deletes a file named FILE1
7. Look at file-list area Select [A:] in drive-icons area	DIR DIR A:	Lists files in current directory Lists files on floppy disk in drive A:
8. Select file FILE1 in file-list area Select File from menu bar Select "Copy" Type A:	COPY FILE1 A:	Copies the file FILE1 to the the floppy disk in drive A:
9. Select file FILE1 in file-list area Select File from menu bar Select "Rename" Type FILE2	REN FILE1 FILE2	Renames the file FILE1 as FILE2
10. Select file FILE1 in file-list area Select File from menu bar Select "Move" Type A:	MOVE FILE1 A:	Moves the file FILE1 from current directory to the floppy disk in drive A:

RANDOM HOUSE
ELECTRONIC PUBLISHING

FRIENDLY COMPUTER BOOKS

A Quick, No-Nonsense Approach to Learning the Basics of DOS 6

As a first-time user, you need to learn to use DOS 6 to do your work more productively. You're not interested in lengthy explanations. Instead, you want an introduction that teaches you the basic features of the software, gets you up and running quickly, and lets you do your work.

That's why we've created Friendly Computer Books. The topic-oriented organization, spacious design, step-by-step approach, and liberal use of illustrations ensure that the essential details needed to use the software literally jump out at you. The special lay-flat binding guarantees that you won't have to fight with your book as you read it. And the very low price means that learning DOS 6 does not require a major investment.

Since the publication of Bantam's *Windows 3 Power Tools*, The LeBlond Group has developed a reputation as the premier authors of books on operating systems.

0-679-79190-6
U.S. $6.99
Canada $7.99

NOVICE

PC

OPERATING SYSTEMS

ISBN 0-679-79190-6

9 780679 791904 56990